AD-A103 000 AIR FORCE GEOPHYSICS LAB HANSCOM AFB MA F/G 4/2
A DIGITAL PROCESSING AND DISPLAY SYSTEM FOR THE ROTATING BEAM C--ETC(U)
FEB 81 J C WEYMAN, R H LYNCH
UNCLASSIFIED AFGL-TR-81-0027 NL

1 OF 1
AD
A103000

END
DATE
FILMED
9-81
DTIC

AFGL-TR-81-0027
INSTRUMENTATION PAPERS, NO. 299

LEVEL II

A Digital Processing and Display System for the Rotating Beam Ceilometer (AN/GMQ-13)

JAMES C. WEYMAN, Capt, USAF
RICHARD H. LYNCH

4 February 1981

Approved for public release; distribution unlimited.

METEOROLOGY DIVISION PROJECT 6670
AIR FORCE GEOPHYSICS LABORATORY
HANSCOM AFB, MASSACHUSETTS 01731

AIR FORCE SYSTEMS COMMAND, USAF

This report has been reviewed by the ESD Information Office (OI) and is releasable to the National Technical Information Service (NTIS).

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

John Howard

Chief Scientist

Qualified requestors may obtain additional copies from the Defense Technical Information Center. All others should apply to the National Technical Information Service.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: AFGL-TR-81-0027	2. GOVT ACCESSION NO.: AD-A103000	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): A DIGITAL PROCESSING AND DISPLAY SYSTEM FOR THE ROTATING BEAM CEILOMETER (AN/GMQ-13)		5. TYPE OF REPORT & PERIOD COVERED: Scientific. Interim.
		6. PERFORMING ORG. REPORT NUMBER: IP No. 299
7. AUTHOR(s): James C. Weyman, Capt, USAF; Richard H. Lynch		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Air Force Geophysics Laboratory (LYU) Hanscom AFB Massachusetts 01731		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62101F 66701007
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Geophysics Laboratory (LYU) Hanscom AFB Massachusetts 01731		12. REPORT DATE: 4 February 1981
		13. NUMBER OF PAGES: 68
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report): Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Rotating Beam Ceilometer
Processor/display
Cloud base height
Microprocessor subsystems
Interrupt-driven
Binomial smoothing function

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A microprocessor-based automated display system for the Rotating Beam Ceilometer (RBC) was developed to demonstrate the feasibility of modernizing and improving the present display system. The new system consists of a Burroughs self-scan panel display (model number SSD0124-0039) and an Intel 8080A-based microcomputer. The program acquires the signal returns from the RBC, converts the analog signal to a digital value, and displays the most recent scan. The system allows for three modes of operation or processing.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract (Continued)

Two modes, a one-scan mode and a five-scan mode, require no further inputs from the user once they have been selected. A manual mode which is user-interactive is also available. The basic display capabilities consist of a depiction of signal intensity vs. height, a numerical display of the height of the peak value and an illuminated cursor positioned at the peak value. Evaluation of the system at AFGL and Scott AFB, IL has confirmed that the display is more readable, provides more accurate cloud heights and permits better interpretation of the data obtained than does the current CRT display system. Further considerations which will be incorporated before operational implementation are discussed.

Preface

The successful fabrication and testing of the processor/display for the Rotating Beam Ceilometer (RBC) could not have been achieved without the technical support provided by TSgt James Boyce. In addition, the authors are grateful to Mr. William Lamkin for his assistance in the fabrication of the equipment, to Mr. Donald A. Chisholm and Lt. Col. William Wright, HQ AWS, for valuable discussions concerning the project, and to Mr. Chisholm for his many helpful comments on the paper. Lastly, the contribution of Miss Karen Sullivan in typing the manuscript is gratefully acknowledged.

DISCLAIMER

The United States Air Force does not approve, recommend, or endorse any product except for its own use, and naming an item is for the purpose of identification only and shall not be used in advertising, sales promotion, or to indicate in any manner, either implicitly or explicitly, endorsement by the U.S. Air Force.

Contents

Illustrations

A Digital Processing and Display System for the Rotating Beam Ceilometer (AN/GMQ-13)

1. INTRODUCTION

The Rotating Beam Ceilometer (RBC) is one of the elder statesmen in the Air Weather Service's inventory of sensors. As with many complex sensors that have been in service an extended period of time, maintenance problems (decreased mean time between failures, availability of parts, etc.) become aggravated. This has been true of the RBC, particularly its display portion. In response to this need, the Air Force Geophysics Laboratory (AFGL) undertook the development of a new and improved display system.

An exploratory development program was initiated at AFGL to design, fabricate, test, and evaluate an automated cloud display system which relied on microprocessor technology. This system was designed to be a low cost, compact package which could provide sufficient detailed information to the operator so that he could rapidly make an accurate observation. The developed display accomplishes this goal. In addition, the microprocessor-based system analyzes the data and produces an objectively determined cloud base height from the five most recent scans. This objective cloud base height could be helpful to a busy observer, but more importantly could be transmitted through a serial channel as part of the

(Received for publication 4 February 1981)

automated observation (AUTO OBS) portion of the Automated Weather Distribution System (AWDS).[1]

This report will review the specific design of the RBC, the microprocessor-based display system, the data received from the RBC, how that data are processed and displayed, and future considerations for the system.

2. EQUIPMENT

2.1 AN/GMQ-13 Rotating Beam Ceilometer (RBC)

The standard RBC system consists of a projector, a detector, and a CRT indicator as shown in Figure 1. The projector contains two back-to-back lamp assemblies on a motor driven mount which continuously rotates in the vertical plane. These lamp assemblies each contain a high current incandescent lamp inside a rotating shutter and a 24 in. parabolic mirror behind the lamp. As the mount rotates, each lamp will project a 120 Hz modulated beam which sweeps from 0 to 90 deg or from parallel to perpendicular to the ground. During the 0 to 90 deg sweep a filter on the housing attenuates visible light allowing infrared light to be transmitted. At 90 deg, the beam is reduced in power and blocked by the housing during the remaining three quadrants of mount revolution. The two-lamp arrangement doubles the atmospheric illumination portion of each revolution. This produces more frequently updated information and hardware redundancy but can introduce problems, discussed later, when computer-processing these signals. Magnets actuate a reed switch when either lamp is horizontal (at 0 deg) to develop a synchronizing pulse for the indicator.

The detector consists of a vertically viewing, parabolic mirror optical system with a photocell at its focal point. A honeycomb collimator is mounted above the mirror to reduce effects caused by sunlight and other stray light. An amplifier boosts photocell output for transmission to the indicator. A locally-fabricated reflector is usually mounted on the housing to reflect some light directly into the optical system when illuminated by a projector's lamp in the 0 deg or horizontal position. This produces a strong return on the indicator and is used to verify system operation. This feature of the system is used by the microcomputer and will be discussed later.

The operational RBC system output is displayed on a cathode ray tube (CRT) indicator. The switch closure which occurs when either lamp is at 0 deg is used to synchronize the sweep mechanism. The electron beam is swept from the bottom to the top of the CRT in synchronization with the projector's rotation from

1. Air Force Communications Service (1977) Automated Weather Distribution System (AWDS) Required Operational Capability (AFSC ROC 601-77).

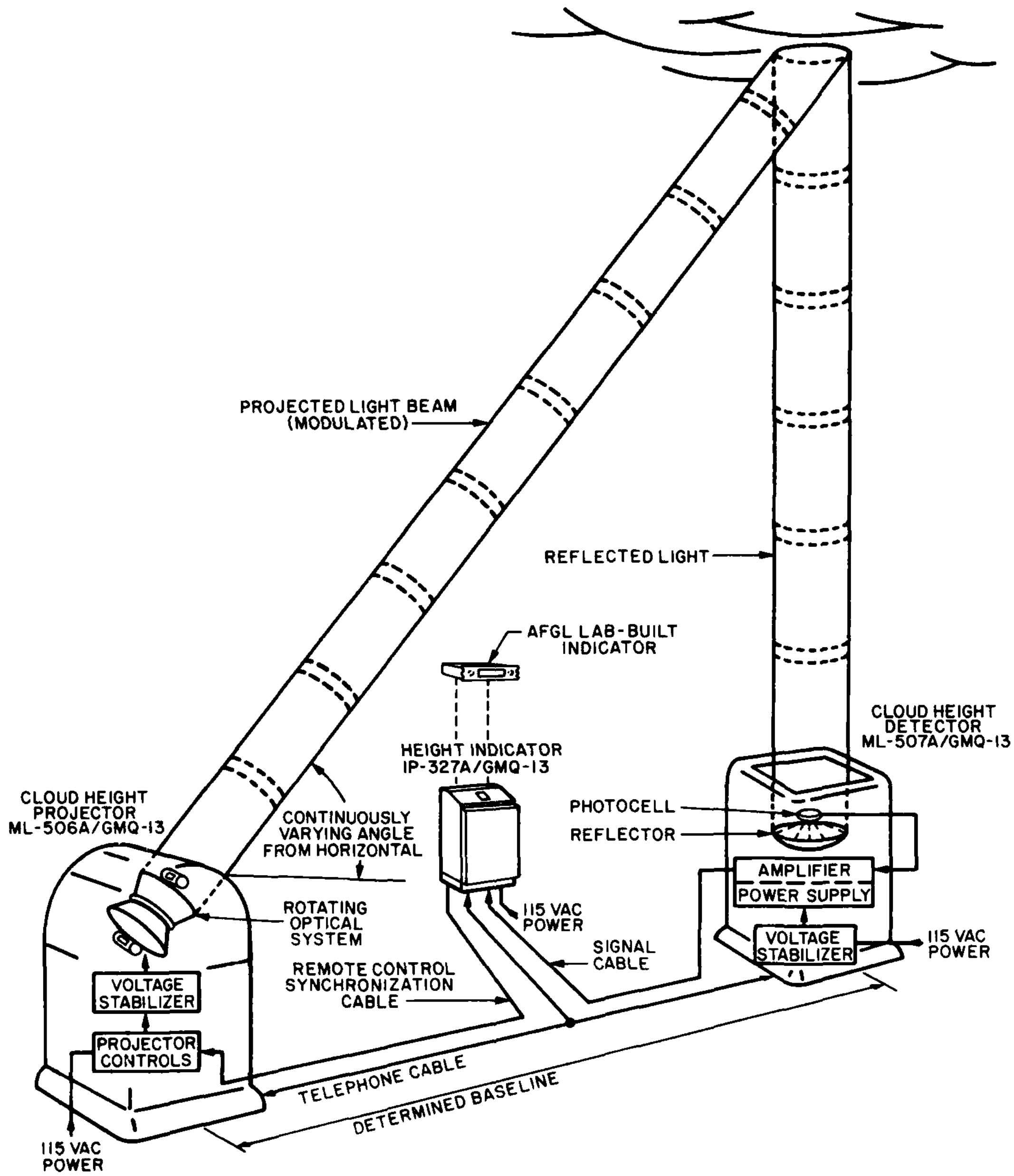


Figure 1. System Diagram

0 to 90 deg. The deflection of the beam left and right of the centerline is proportional to signal strength; that is, a strong cloud return gives a wide deflection. Operation of the indicator often requires numerous adjustments and considerable operator skill in interpreting signals resulting from multiple cloud layers, snow, fog, etc.

Actual cloud height is determined by watching the trace for the 3 sec it takes to scan from bottom to top of the CRT. If a cloud return is present, the point of maximum beam deflection is taken to be the height of the cloud's base. This is read directly from an overlay indicating height vs. projector angle or by reading an angle-only overlay and referring to a conversion table to obtain the cloud height. Numerous operator adjustments must be correctly done to maintain system accuracy. In addition, synchronization with the projector must be manually initiated.

2.2 AFGL RBC Processor/Display

The AFGL RBC processor/display unit (see Figure 2) consists of the unit's front panel and five functionally separate subsystems contained on three printed circuit cards. The five subsystems are:

1. Analog demodulator and converter,
2. Microcomputer,
3. Dot matrix display,
4. Display controller,
5. Power supply.

The analog demodulator and converter circuit is shown schematically in Figure 3. This circuit accepts the 120 Hz amplitude modulated signal from the RBC. An absolute value rectifier and integrator (M31) detects and slightly filters the envelope of the signal which is amplified by the programmable gain differential amplifier (M32). Gains of 1.00, 1.33, 1.67 and 2.00 are available under program control by means of a two pole, solid state analog switch (M30). This switch is driven by two computer output lines and alters the amplifier's feedback resistance. In addition, a dc voltage is applied to the amplifier's differential input in order to offset the residual dc baseline voltage. The offset voltage can be one of 16 values developed by a digital to analog converter (M30) and ranges from near zero to approximately 25 percent of the full scale 10 V input. The firmware description in this report (see Section 4) details the methods used to determine gain and offset values based on analysis of the signal. An 8 bit analog to digital converter (M28) completes the processing of the signal. A programmable input/output port (M27) serves as the interface between the analog and computer subsystems.

The microcomputer in this system is based on an Intel 8080A microprocessor and is shown schematically in Figure 4. Program memory capacity is 12K of 2716 type EPROM while random access memory capacity is 4K of 8108 static RAM.

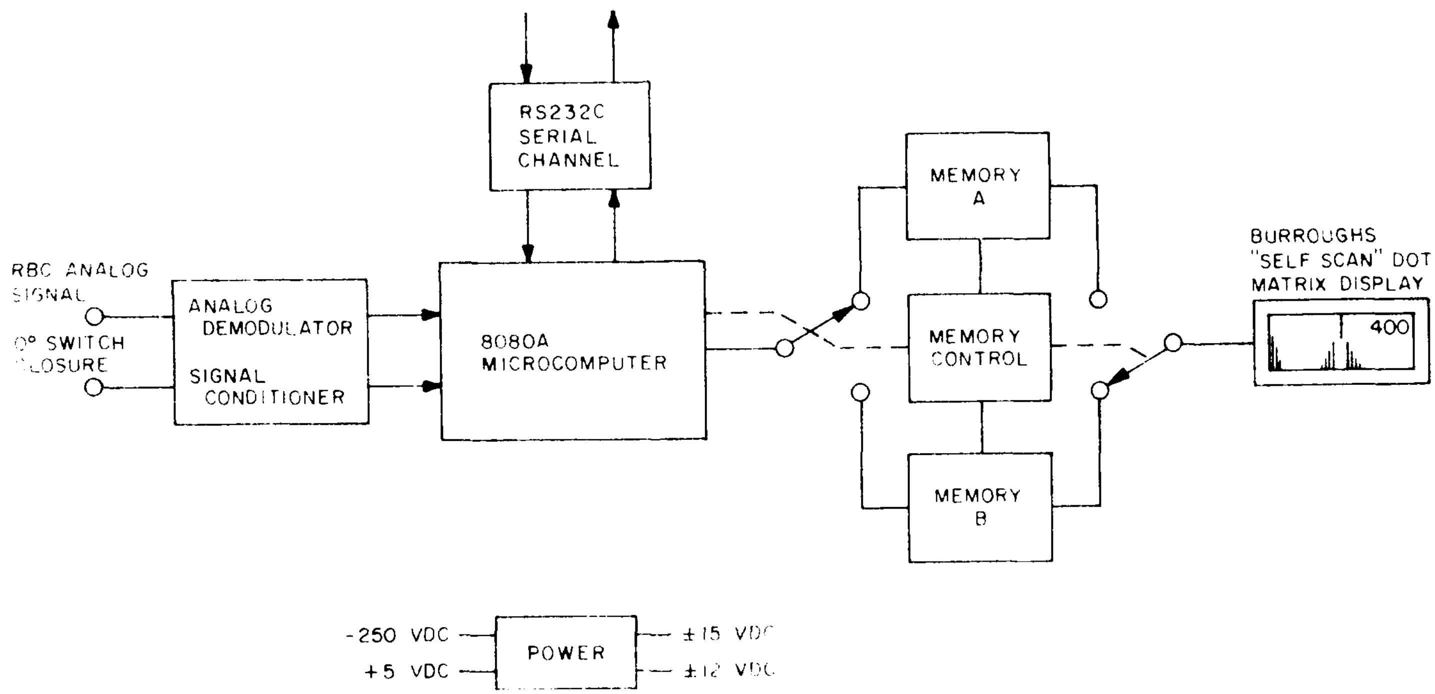
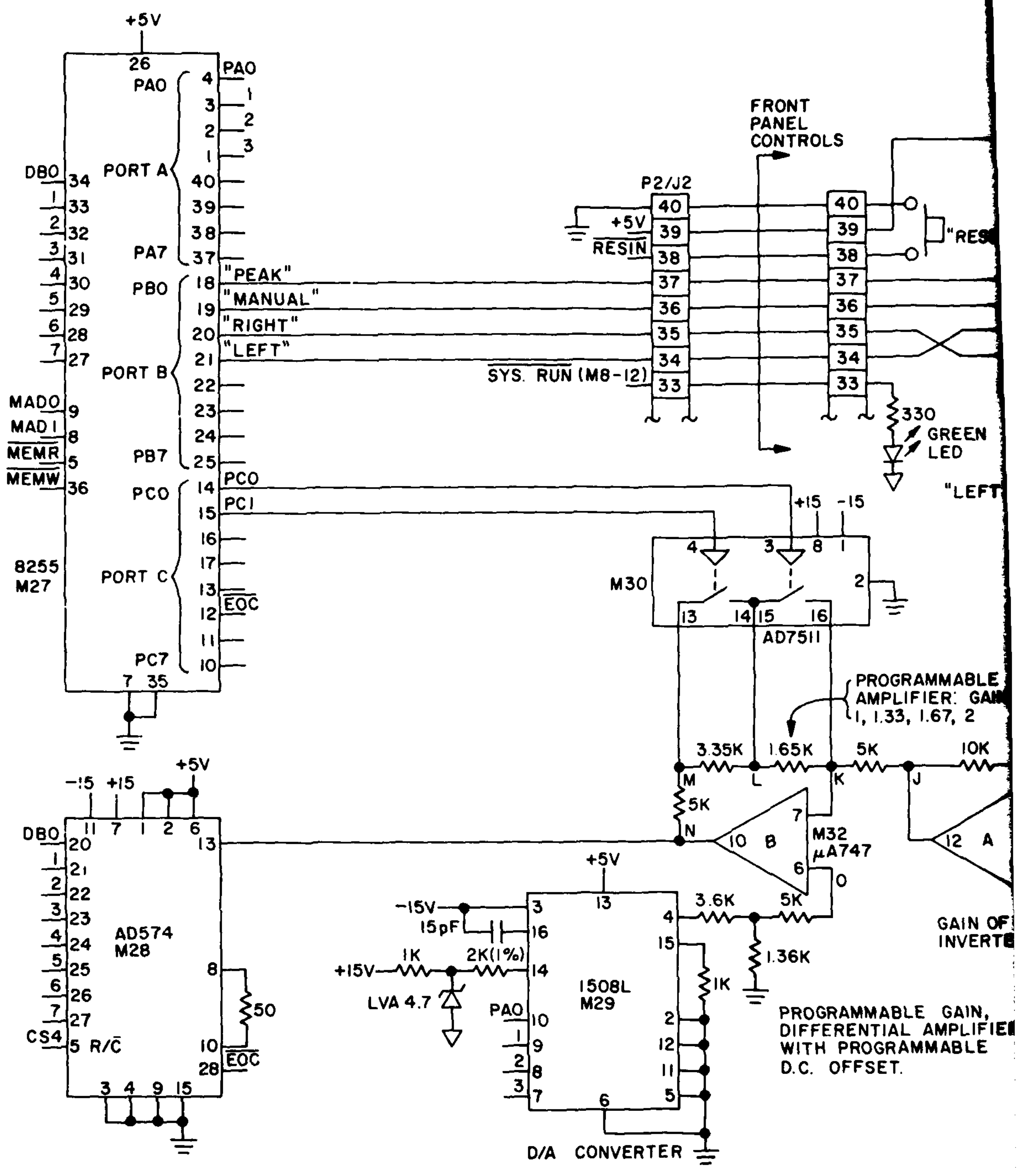
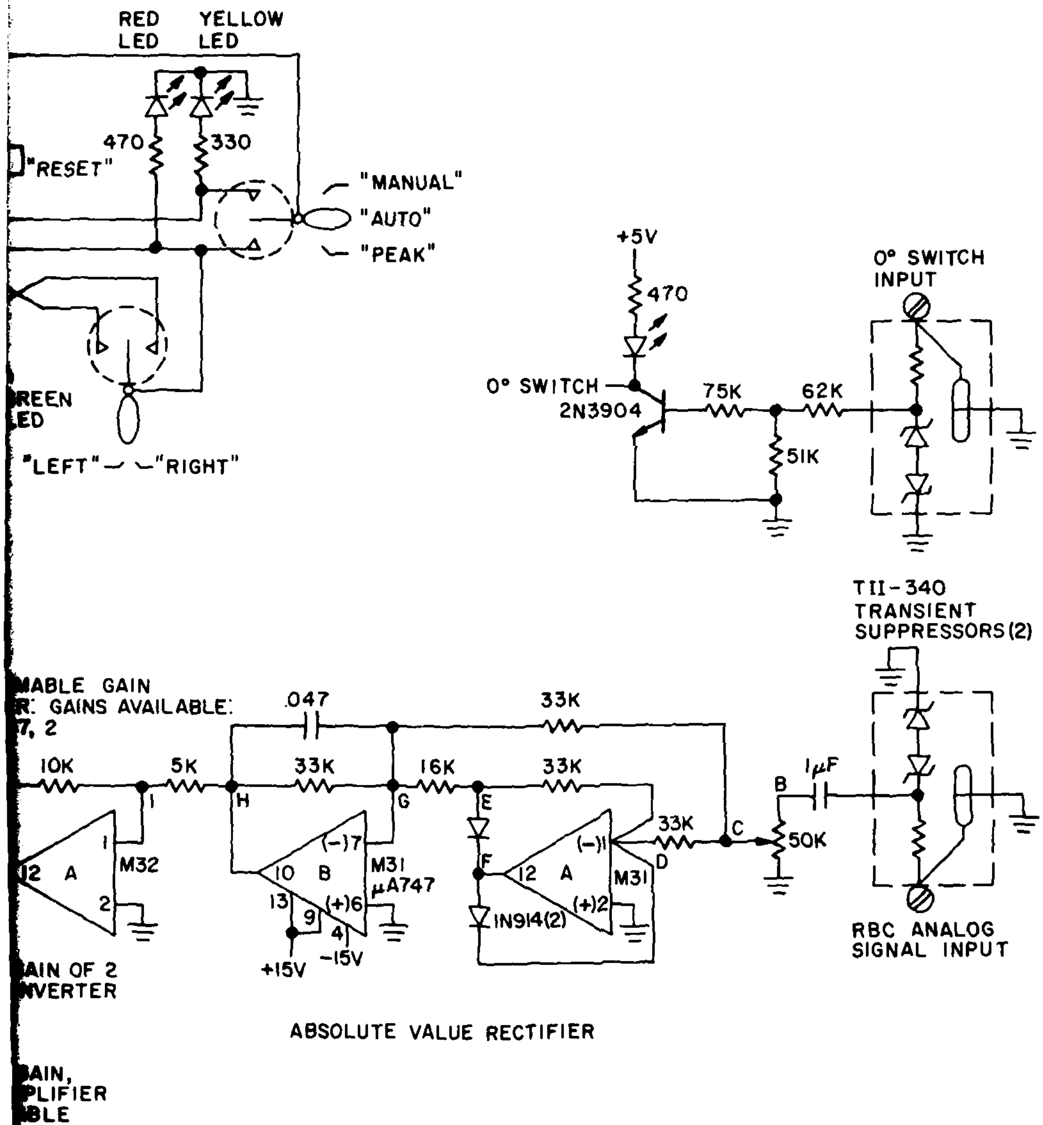


Figure 2   RBC Display Processer (Block Diagram)



+5V
8255
M27
PORT A
PORT B
PORT C
PA0
PA7
PB0
PB7
PC0
PC1
PC7
DB0
MAD0
MAD1
MEMR
MEMW
"PEAK"
"MANUAL"
"RIGHT"
"LEFT"
EOC
FRONT
PANEL
CONTROLS
P2/J2
RESIN
SYS. RUN (M8-12)
330
GREEN
LED
"RES
"LEFT
+15
-15
M30
AD7511
PROGRAMMABLE
AMPLIFIER: GAI
1, 1.33, 1.67, 2
3.35K
1.65K
5K
10K
M32
μA747
3.6K
1.36K
GAIN OF
INVERTE
AD574
M28
R/C
CS4
50
-15V
15pF
+15V
1K
2K(1%)
LVA 4.7
1508L
M29
D/A CONVERTER
PROGRAMMABLE GAIN,
DIFFERENTIAL AMPLIFIE
WITH PROGRAMMABLE
D.C. OFFSET.



THIS CIRCUITRY IS LOCATED ON DISPLAY CONTROLLER BOARD (EXCEPT AS NOTED)

Figure 3. Analog Demodulator and Signal Conditioner (Schematic Diagram)

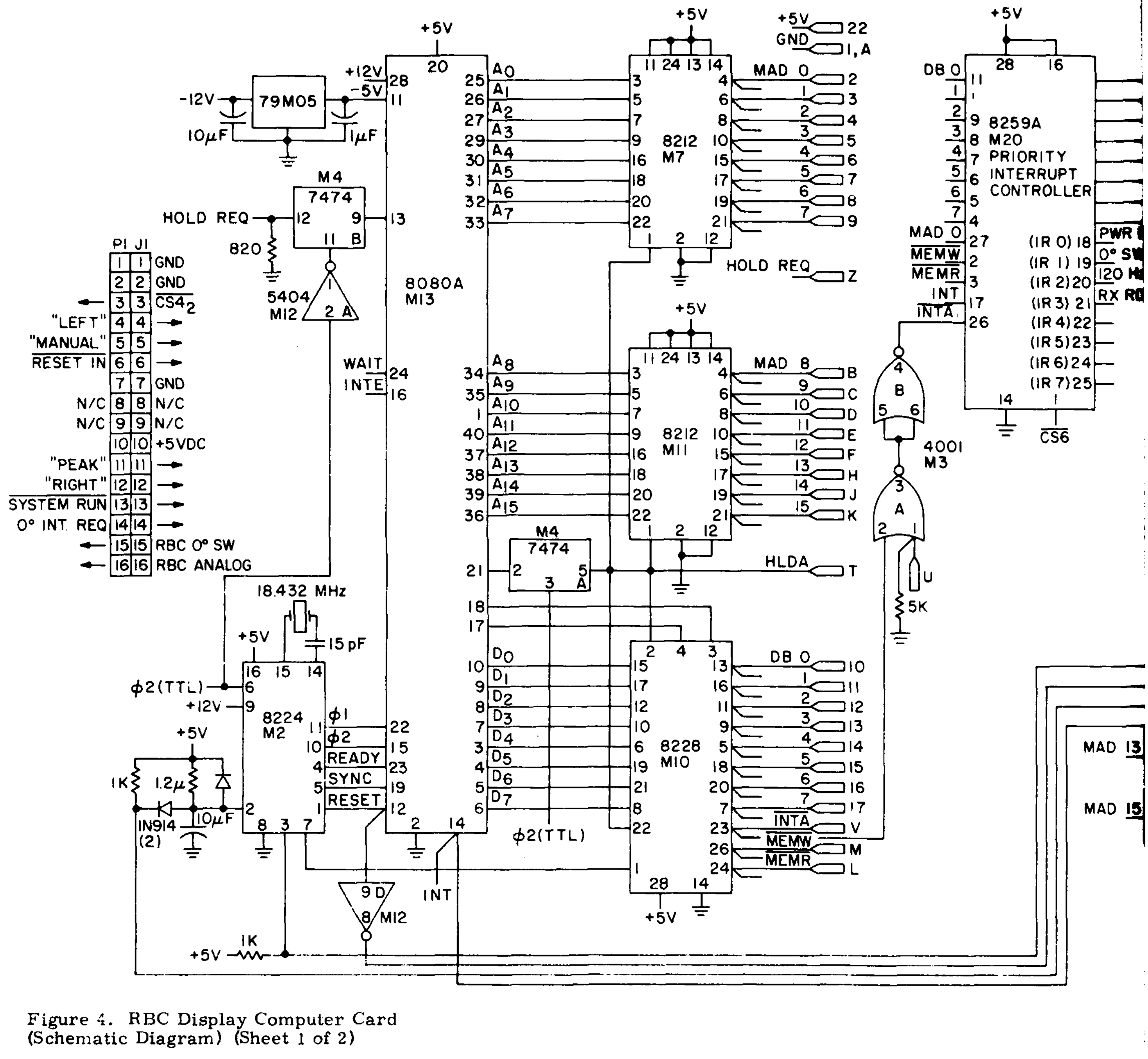
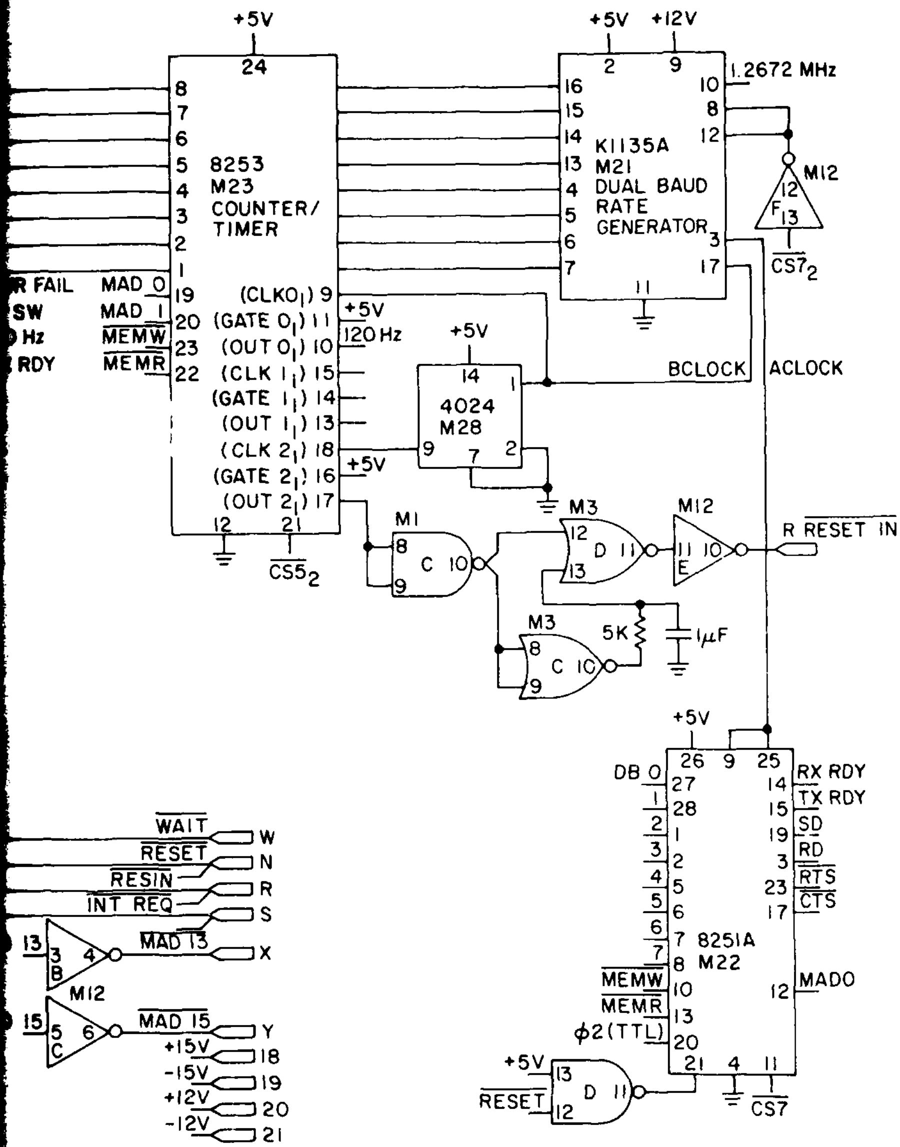


Figure 4. RBC Display Computer Card
(Schematic Diagram) (Sheet 1 of 2)



+5V
8253
M23
COUNTER/
TIMER
+12V
1.2672 MHz
K1135A
M21
DUAL BAUD
RATE
GENERATOR
M12
CS7
R FAIL
SW
Hz
RDY
MAD 0
MAD 1
MEMW
MEMR
(CLK0)
(GATE 0)
(OUT 0)
(CLK 1)
(GATE 1)
(OUT 1)
(CLK 2)
(GATE 2)
(OUT 2)
120 Hz
4024
M28
BCLOCK
ACLOCK
CS5
M1
M3
R RESET IN
5K
1μF
8251A
M22
DB 0
RX RDY
TX RDY
SD
RD
RTS
CTS
MADO
φ2(TTL)
RESET
WAIT
RESIN
INT REQ
MAD 13
MAD 15
+15V
-15V
+12V
-12V

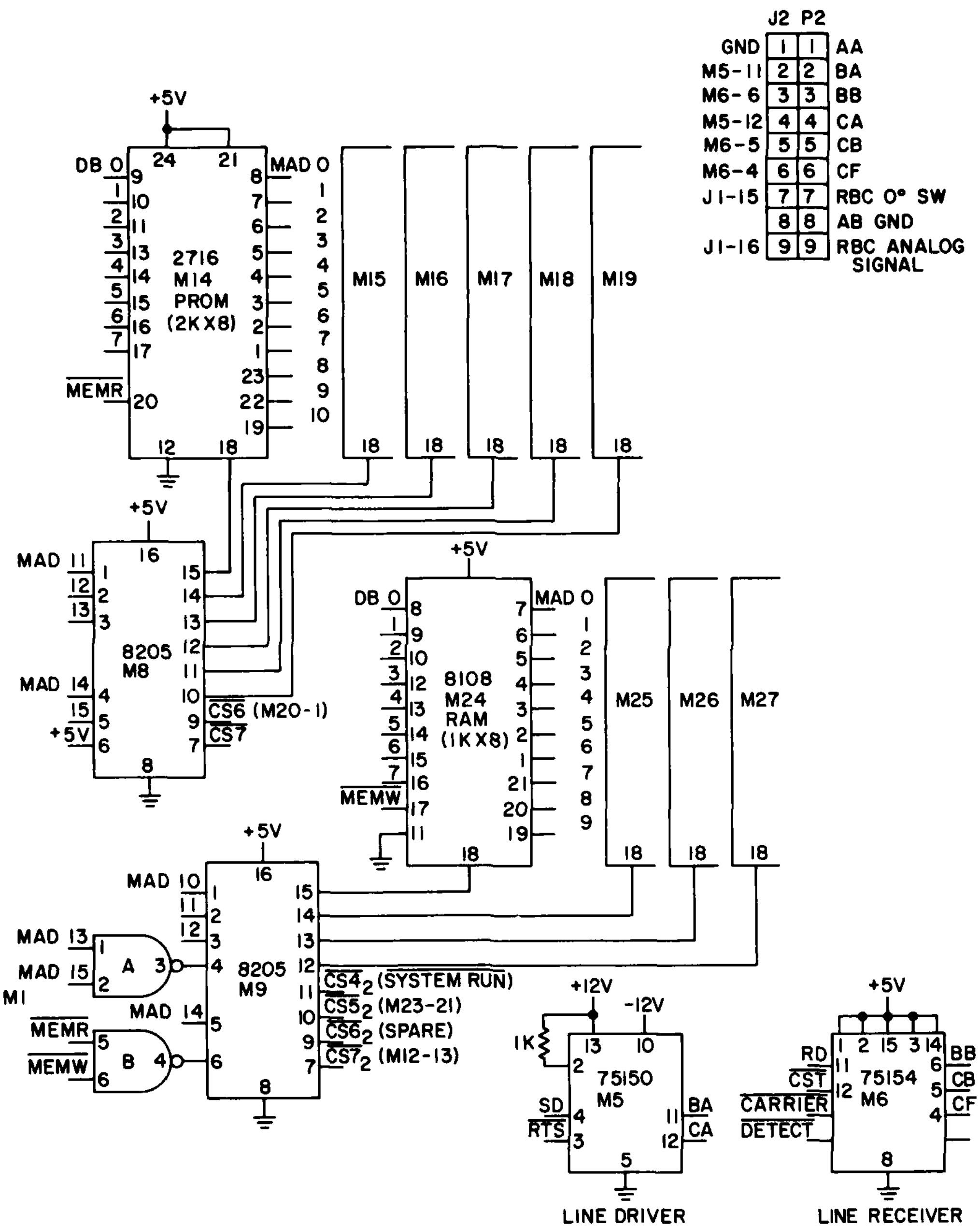


Figure 4. RBC Display Computer Card (Schematic Diagram) (Sheet 2 of 2)

The system incorporates an eight level priority interrupt controller (M20), a triple 16 bit counter/timer (M23), a USAR/T serial port (M22) with RS 232C interface and a programmable baud rate generator (M21). This combination of devices, contained on one board, has been found to be a very useful "workhorse" microcomputer in earlier lab designs and provides considerable computer power and versatility in this application.

The dot matrix display includes a Burrough's SSD-0124-0039 panel and driver assembly.[2] The assembly consists of a SELF-SCAN panel display and driver board. The SELF-SCAN panel display is a flat panel gas discharge display device capable of displaying a matrix of 17 cells by 192 cells which can be configured in any font or graphic application. The cells are located on 0.040 in. centers vertically and 0.030 in. centers horizontally.

The driver board consists of a 3-phase counter, anode drivers, phase drivers, missing pulse detector, blanking control, and scan anode disable control circuit. The display clock signal is used to sequentially address the cathode phases. The data input signals are used to address the display cells at the appropriate location. The driver board contains a protection circuit. If loss of the clock or reset signal is detected, it switches off the display and scan anodes to protect the display panel from being damaged while operating in a non-scan mode.

The display operates in a scanning mode, scanning from left to right, one full column at a time. A clock must be included with a period of one column time. At the end of the 192nd display column time, the negative edge of the reset pulse must be generated to initiate a new scan cycle. The reset duration must be at least one clock period.

Seventeen data input lines are provided to address the PNP transistors which are operated in a constant current source configuration to drive the display cells.

The protection circuit consists of two retriggerable monostable multivibrators arranged in a missing pulse detector configuration to monitor the clock and reset input signals. Failure to receive high to low transitions on either line at appropriate time intervals disables the display anodes, scan anodes, and cathode phase drivers. The presence of these signals at a later time would automatically resume the normal operation of the system.

The simplified schematic of the display controller circuit is shown in Figure 5 (Figures 6 and 7 are the detailed schematics). The Burroughs SELF SCAN dot matrix display is organized into 192 sixteen bit vertical columns. A 17th bit is wired true to provide an illuminated baseline.

---

2. Burroughs Bulletin No. 3511A, Burroughs Corporation, OEM Division, P.O. Box 1226, Plainfield, New Jersey 07061.

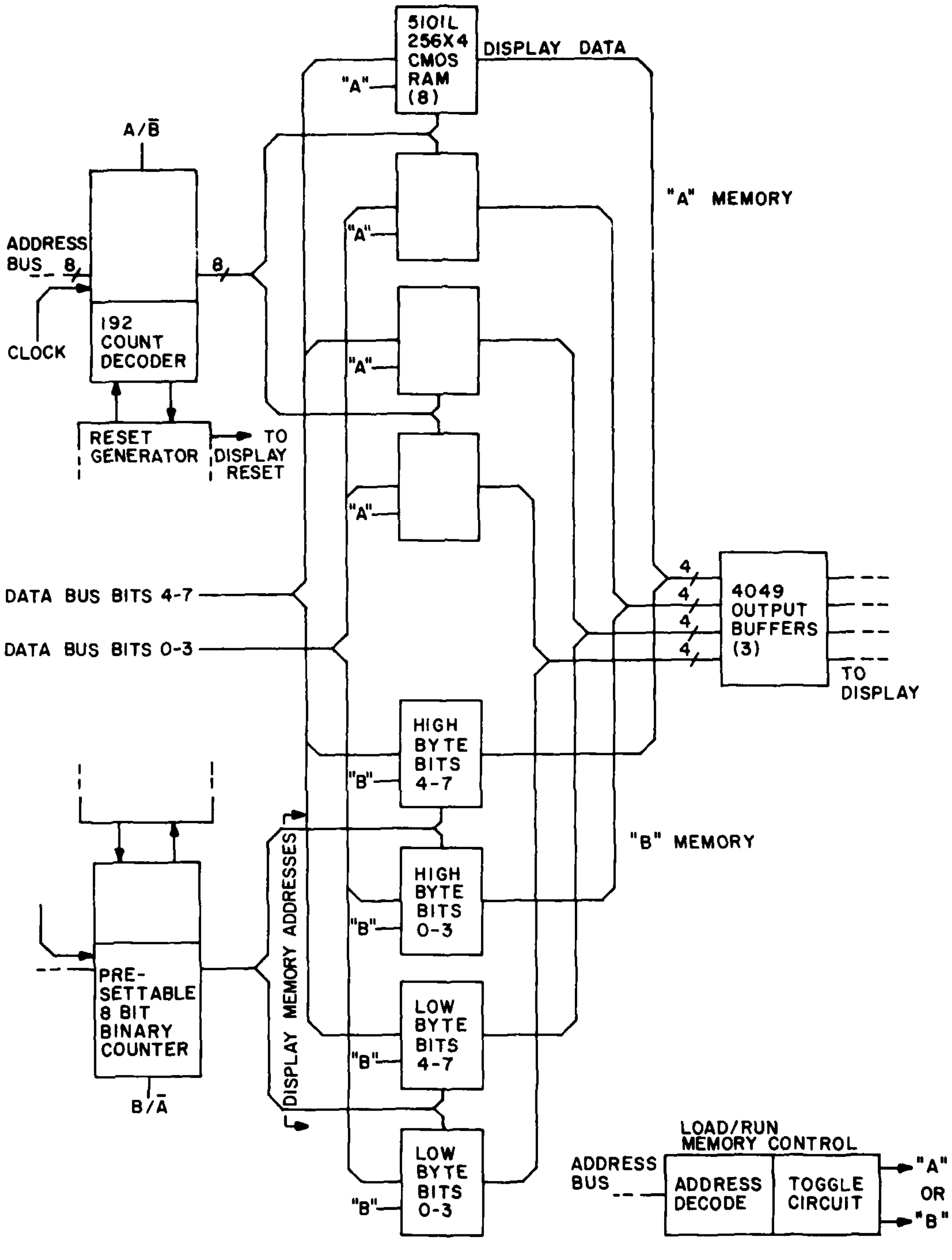
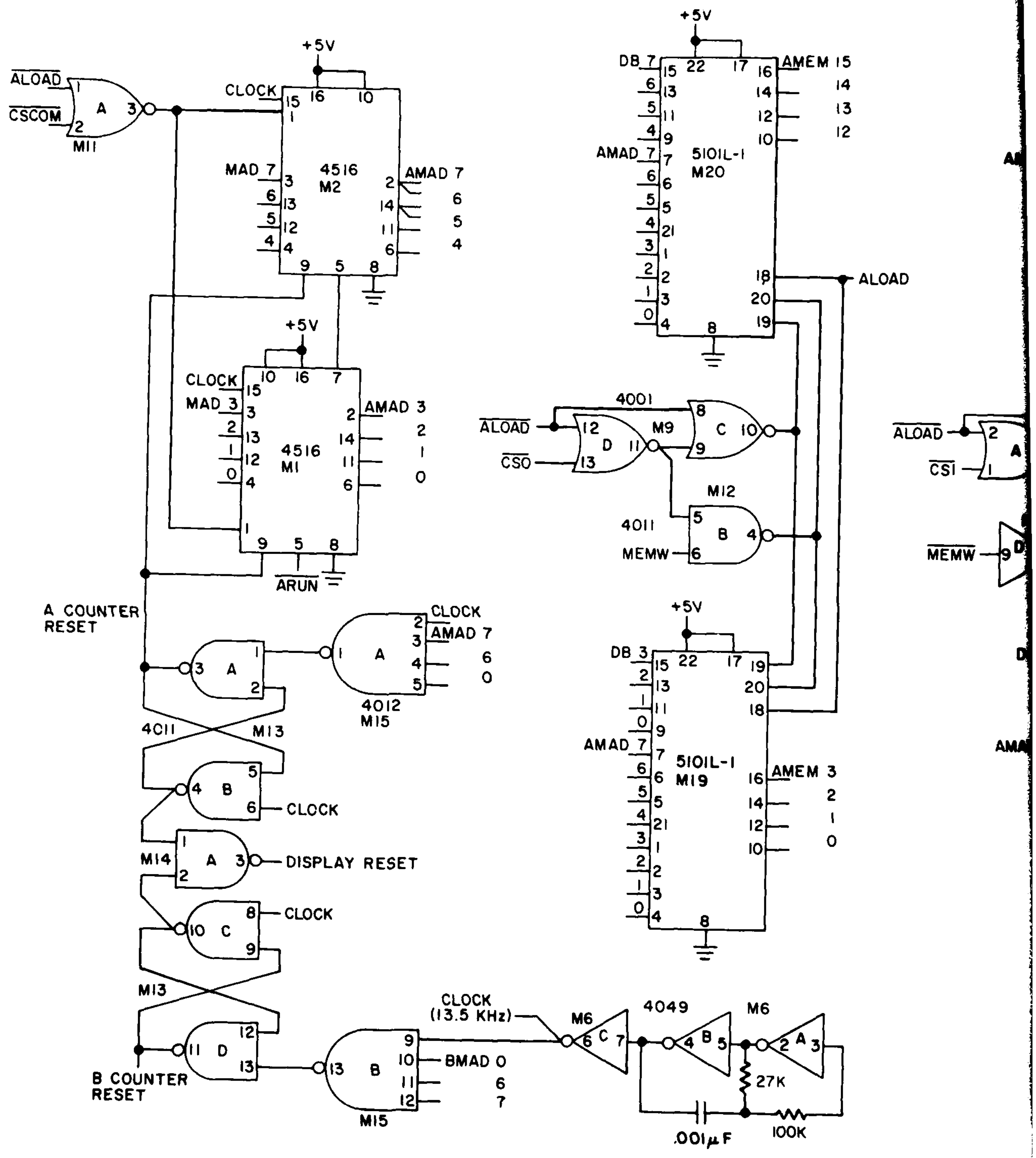


Figure 5. Double Dual-Port Display Memory (Simplified Schematic)



ALOAD
CSCOM
A
M11
+5V
CLOCK
4516
M2
MAD 7
6
5
4
AMAD 7
6
5
4
CLOCK
MAD 3
2
1
0
4516
M1
AMAD 3
2
1
0
ARUN
A COUNTER
RESET
CLOCK
AMAD 7
6
0
4012
M15
4011
M13
CLOCK
M14
DISPLAY RESET
CLOCK
M13
B COUNTER
RESET
BMAD 0
6
7
M15
CLOCK
(13.5 KHz)
M6
4049
M6
27K
.001μF
100K
DB 7
6
5
4
AMAD 7
6
5
4
3
2
1
0
5101L-1
M20
AMEM 15
14
13
12
ALOAD
4001
ALOAD
CSO
M9
M12
4011
MEMW
DB 3
2
1
0
AMAD 7
6
5
4
3
2
1
0
5101L-1
M19
AMEM 3
2
1
0
ALOAD
CS1
MEMW

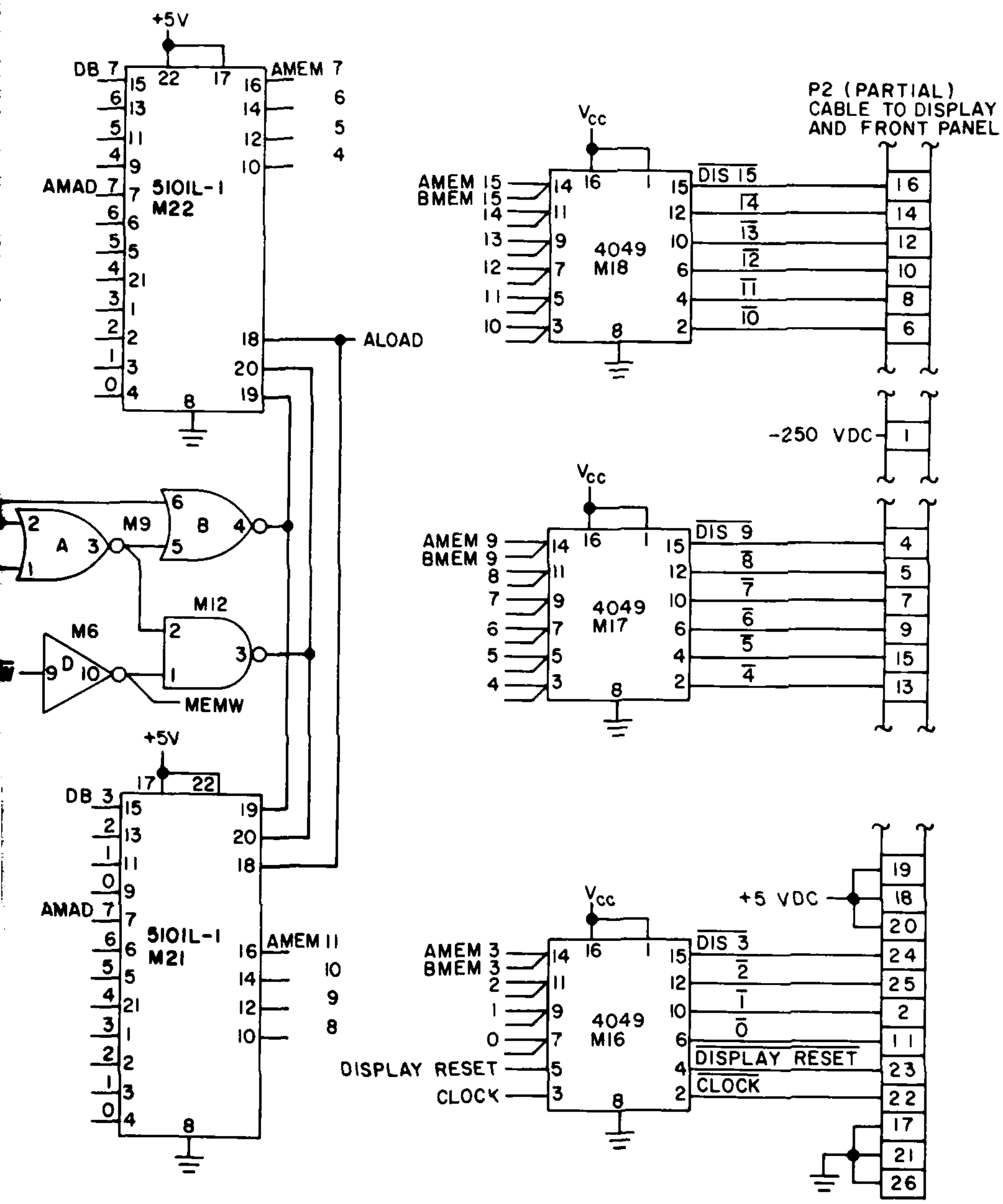


Figure 6. "A" Memory and Display Clock (Schematic Diagram)

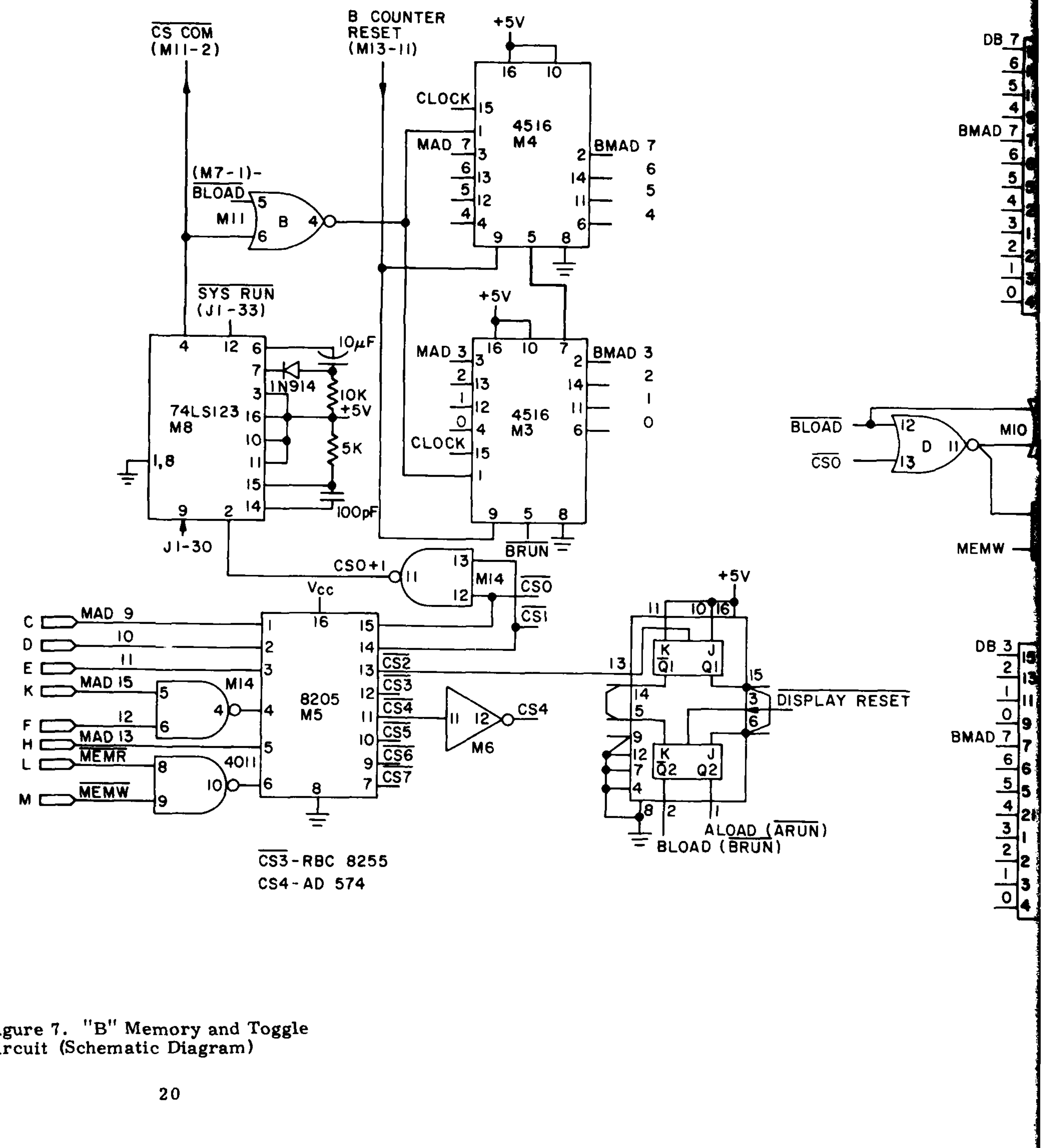
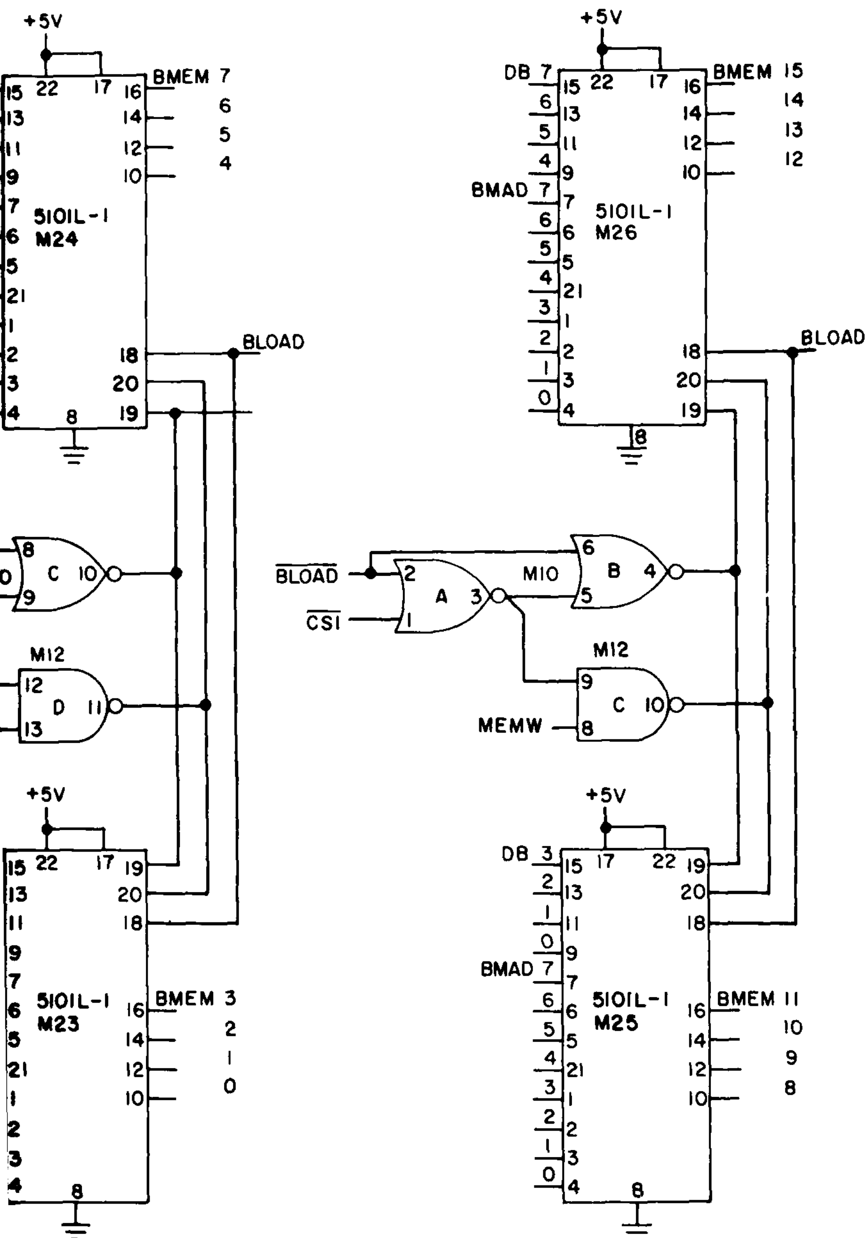


Figure 7. "B" Memory and Toggle Circuit (Schematic Diagram)


+5V
BMEM 7
6
5
4
5101L-1
M24
BLOAD
C
M12
D
+5V
BMEM 3
2
1
0
5101L-1
M23
+5V
DB 7
BMAD 7
BMEM 15
14
13
12
5101L-1
M26
BLOAD
$\overline{BLOAD}$
$\overline{CS1}$
A
M10
B
M12
C
MEMW
+5V
DB 3
BMAD 7
5101L-1
M25
BMEM 11
10
9
8

2

To create an image, a data field is stored in the display memory in the form of 192 low byte/high byte pairs or columns. Approximately 70 times each second an address generator made up of an oscillator (M6) and binary counters (M1 and M2 in A memory or M3 and M4 in B memory) provides sequential addresses and data strobes to write a data field to the display from left to right. The 193rd clock pulse, conditioned by the counter's decoder (M15), develops a Display Reset strobe. If the computer is not requesting a memory "toggle" or update, the memory merely cycles and continues to display the same data. A "toggle" request, stored by M7, will cause a switch to the alternate memory synchronized by Display Reset. Contents of the alternate memory are now clocked to the display in the above manner.

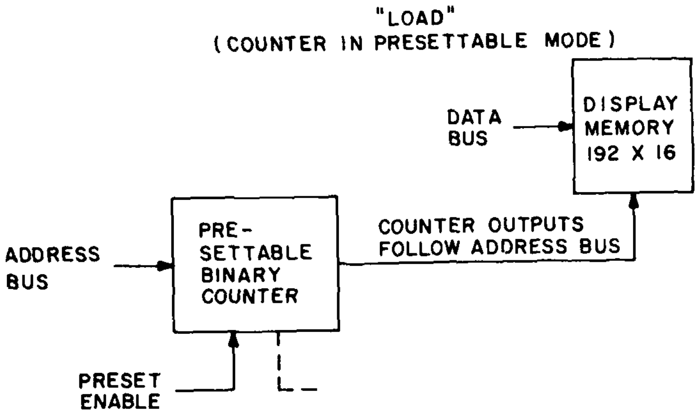
A dual purpose 8 bit presettable binary counter (whose operation is illustrated in Figure 8) develops or buffers display memory addresses depending on memory selection. If in "load" state, the counter is in its presettable condition and merely buffers the lower 8 bits of the computer's address buss directly to the selected display memory. When data is to be written to this memory, the computer uses an ordinary memory-write instruction. When toggled to "run" state, display memory addresses are developed as the counter accumulates column-strobing clock pulses.

A total of seven dc voltages are required to operate this equipment: +5, -5, ±12, ±15, and -250. The +5 Vdc logic power is developed from the 120 V, 60 Hz line by means of a high efficiency, switching type power supply. With the exception of -5 Vdc, the other voltages are developed by dc to dc converters powered by the +5 V supply. The -5 Vdc power for the 8080A is developed from the -12 Vdc source through a three terminal regulator.

3. ROTATING BEAM CEILOMETER DATA

The configuration and operation of the RBC was described in some detail in Section 2.1. Basically, the detector is set a fixed distance away from the projector (normally 400 ft), with its field of view vertical and coplanar with the rotating projector beam. The sensor's intersection volume advances up the detector's vertical beam as the projector's beam rotates from the horizontal. When the volume coincides with a cloud, as illustrated in Figure 9, backscatter of the projector's beam by water droplets in the cloud is detected by the receiver and depicted on the indicator. The elevation angle (α) at which the maximum backscatter return occurs yields the cloud height (h) by triangulation.

It is characteristic for the RBC response to increase slowly above the noise level, reach a maximum, and decrease gradually to the noise level. This is illustrated in Figure 10. Referring to Figures 9 and 10, the return begins to



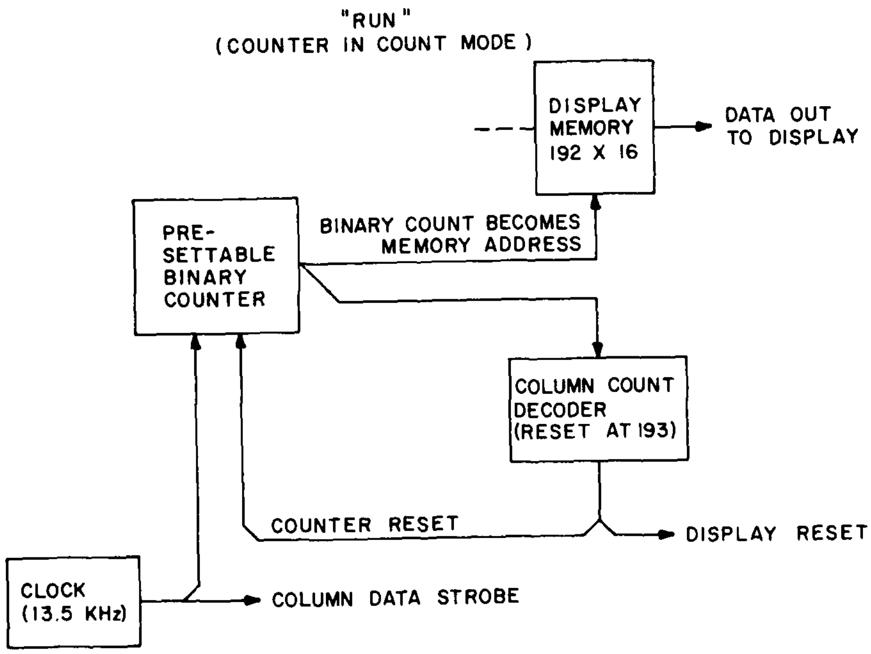


Figure 8. Display Memory Load/Run Detail

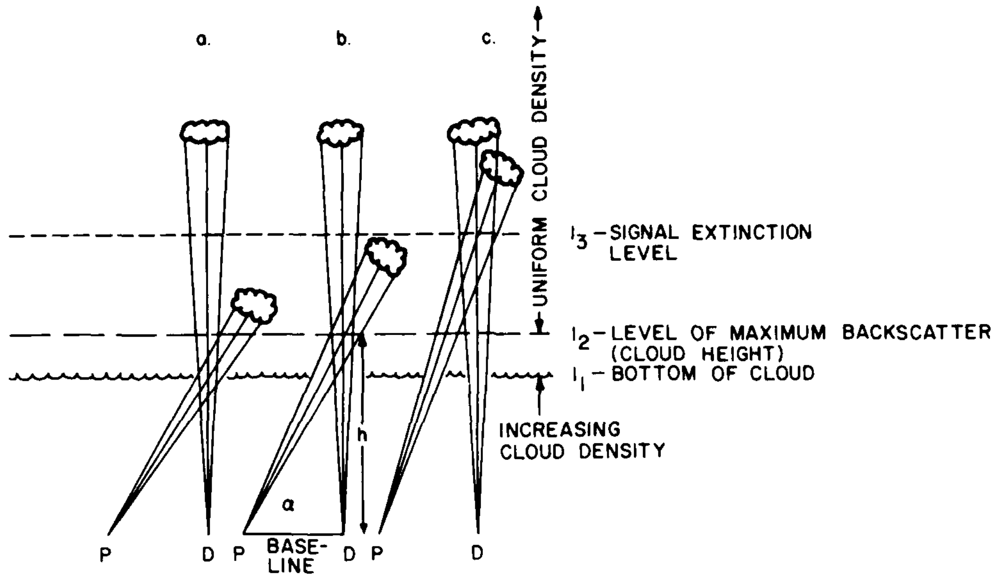


Figure 9. RBC Measurement Geometry

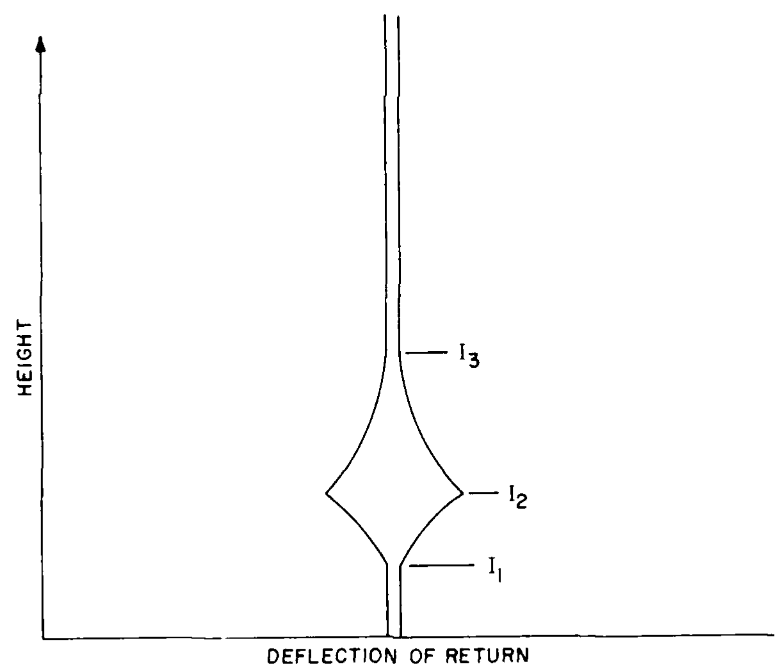


Figure 10. Diagram of Old RBC Display

increase when the optical axes intersect at I_1, the bottom of the cloud. The signal strength reaches a maximum at the level of maximum backscatter I_2, and then decreases to the noise level at the signal extinction level, I_3.

Moroz and Travers[3] have suggested that this idealized return is altered by the effects of multiple scatter. They postulated that sufficient multiple scatter occurs to create signals above noise level which would be detected just before the two beams intersect the cloud. In addition, multiple scatter between the two beams reinforces the receiver's signal, suggesting that the maximum return would occur at a higher altitude than it exists. Quantitative adjustment for multiple scatter effects were not estimatible from their study.

For the purposes of our development and test, cloud heights were determined in accordance with Federal Meteorological Handbook No. 1 guidelines which state, "The base of sky cover aloft is represented by points of maximum reaction (peak return) of the ceilometer to light reflected from the base of layers detected."[4]

4. AUTOMATED DATA PROCESSING PROCEDURES

The computer program developed to handle the RBC processing is shown diagramatically in Appendix A and the PL/M-80 source code is listed in Appendix B. It was written to process the output from a single Rotating Beam Ceilometer (RBC) using an Intel 8080A microprocessor. It has three main objectives. The first is to represent the data received from the RBC on the display system in a manner similar to the operational CRT display. The second is to filter out spurious or insignificant returns. The last objective is to examine the retained data and to determine a representative cloud base height for the single RBC observing point. It should be noted that each of the two lamps in the RBC is processed independently, almost as if they were two separate instruments. This prevents minor differences in one lamp (for example, lamp intensity variation, a slightly different calibration) from influencing the other.

The logic of the program is based upon the acquisition of certain start signals from the RBC and internally generated 120 Hz clock pulses. Within the RBC, each 6 sec scan of information from one lamp has two parts; a 3 sec, 0 to 90 deg cloud base height data scan and a 3 sec, 90 to 180 deg lamp inactive scan. However, for processing purposes, the 180 deg scan is divided into three parts. The first 5 deg

---

3. Moroz, E.Y., and Travers, G.A. (1975) Measurement of Cloud Height, AFCRL-TR-75-0306, AD A015 737.

4. Federal Meteorological Handbook No. 1, Surface Observations, Second Edition (1979), Washington, DC, pp A5-5 to A5-29.

is used to calculate a gain adjustment, 5 to 90 deg is taken as cloud base height data, and 90 to 180 deg is stored to calculate an offset adjustment for the next cycle of the lamp. A zero degree switch closure signal received from the projector is used to initiate data collection.

The program operates on an interrupt-driven basis. This allows the microprocessor to continuously execute its main program, stopping only to service peripheral devices when it is told to do so by the device itself. In effect, the interrupt method provides an external input that directs the processor to complete the instruction currently being executed and fetch a new routine that will service the requesting device. Once this servicing is complete, the processor automatically resumes exactly where it left off. This allows the processor to operate in a very efficient manner. In addition, a priority or level of importance can be placed on each interrupt. This permits a higher priority interrupt to stop the processing of a lower priority interrupt, allowing the higher one to be serviced immediately. In this system the zero degree switch closure is assigned the highest priority followed by the internally generated 120 Hz clock pulse.

When first put into (or returned to) operation, the computer initialization program formats certain component chips and presets certain variables. The system then acquires the first zero degree switch closure to signal the upcoming receipt of data pulses. Once the program has received the first zero degree switch closure, data acquisition is synchronized by the 120 Hz clock. When a 120 Hz clock pulse is received, the main processor is interrupted and a data acquisition sequence is accomplished. This produces data samples every 8.3 msec or 0.25 degree of elevation, thereby dividing the 180 deg scan into 720 data samples.

Two corrections, obtained from previous scans of the same lamp, are applied to each received data scan. The first, a gain correction applied to pulses 20-719 (5 to 179.75 deg), is determined for each lamp from its previous scan. During the initial segment of the scan a strong signal return is developed by a small reflector (mounted on the detector housing) which is illuminated by the projector beam. The signal reaches a maximum value near zero degrees elevation and decays to a background level of 0.3 to 0.5 V before reaching 5 deg of elevation. This signal approximates a strong cloud base return and is used as a reference for the gain correction. This correction, which compensates for degraded signal intensity, is the factor needed to adjust the signal level that occurs in the first 5 deg, to a value representing a full intensity lamp. Four discrete gain adjustments are permitted: 1.00, 1.33, 1.67, and 2.00. The second correction, an offset applied to pulses 20 to 360 (5 to 90 deg), is determined from a 10-scan running average of the noise or background level of the detector. It is determined during the 3 sec portion of each sweep when the projector beam does not intercept the detector beam (pulses

361 to 719 covering the range 90.25 to 179.75 deg). A simple average of the low-level noise (background) peaks received during this part of the scan is integrated into a 10-scan running mean (which spans 2 min) to provide the offset factor for the next rotation of the same lamp. For this purpose, a noise peak is defined as an increase in the amplitude of the return signal followed by a decrease. Offset factors range from near 0 to 2.35 V in 16 discrete and equal intervals. After one completes the 3 sec lamp inactive period, the other lamp trips the zero degree switch, becoming the active lamp, and the processing repeats itself alternating back and forth from one lamp to the other. The primary data acquisition phase of the program is completed with the switch-over to the other lamp.

One additional piece of preprocessing should be noted. During cloudless daylight periods varying amounts of sunlight may, on occasion, be scattered into the detector depending on time of day and orientation of RBC. If these returns occur during the cloud base data segment (0 to 90 deg), erroneous cloud base heights may be reported. Because these returns are produced by sunlight and not the RBC lamp, they are just as likely to occur during the 3 sec inactive scan of the lamp. Therefore, if two or more peaks collected during the inactive scan have magnitudes of 25 percent or more of full scale, sunglint is assumed and a cloud base height is reported as none. Two or more peaks are used to reduce the possibility that other random noise sources, which may produce one large noise peak, are not misinterpreted as sunlight.

Once the 360th data pulse has been collected, the program automatically begins processing the cloud height data scan. While this processing is occurring, the program, through the interrupt method discussed before, continues to gather data (from pulses 361 to 719) to be used to calculate the offset correction. The first processing step is to prepare the data for display. The maximum return in the first 20 pulses (5 deg) is located. If the zero degree switch is perfectly positioned, this maximum return should occur at the first pulse (zero degrees). If this is not the case, the program shifts the largest return into the zero degree position. This self-calibrating procedure ensures a correct height being determined and prevents inconsistency between the height reported by two lamps which may not have been calibrated in exactly the same matter.

The geometry of the RBC and experience with its performance suggest clouds above 87.5 deg would not scatter sufficient light back into the detector to be discernable with confidence. Therefore, subsequent processing is restricted to pulses 0 to 352 (0 to 87.5 deg). The next step is to reduce the range of pulses (0 to 352) and the range of signal intensity (0 to 255) to fit the array set aside for the 192 by 17 dot matrix display. Selecting every other pulse reduces its effective range to 0 to 176. Dividing the signal intensity by 16 reduces its range to 0 to 15. Thus, if original pulse number 36 had a signal intensity of 158, then the 18th element of the display array would have bits 0 to 8 lit and bits 9 to 15 unlit. (Internally the program would assign a 1 bit to array dots on the display to be lit.) The last portion of this pass through the data is to locate the maximum return within the pulse range 20 to 352. A cursor is displayed at the element of the array which represents this maximum return. For this element of the display array, the lit/unlit logic is reversed. Thus, in the example above, bits 0 to 8 in element 18 would be unlit and bits 9 to 15 would be lit. If the magnitude of all data points in the range 20 to 352 was zero, the cursor would not be displayed.

The system allows for three modes of operation or processing which will be described in subsequent paragraphs. They are the one-scan and the five-scan modes (which require no further inputs from the user once they have been selected) and the manual mode (which is user-interactive). These options are exercised by the user/operator through a switch on the display's front panel. Switching from one mode to another, as required to satisfy the operator's data needs, is easily handled by the display's microprocessor.

The full range data array (0 to 352 pulses and 0 to 255 signal intensity) is filtered and analyzed to find a cloud base height. The first step in this stage involves the detection and rejection of large amplitude single pulse noise spikes. Any pulse (from numbers 21 to 351) whose magnitude is greater than the magnitude of the average of its two adjacent neighbors by 200 percent or more is rejected and replaced by the average of the adjacent values. Then a binomial function is applied to eliminate high frequency components whose duration is less than four pulses. Ripples of less than four pulse duration which are part of substantially higher peaks will also be filtered out. However, since the major peaks persist over a large number of pulses, these will not be eliminated. In this process the absolute location of slope reversal (which identifies each peak) will be preserved. A 7 weight filter is used in this binomial function with the following weights: 1/64, 6/64, 15/64, 20/64, 15/64, 6/64, 1/64. The filter weights are successively applied to the set of pulses, using each pulse as the central value in turn and adjacent pulses to define the wings of the function. The 7 point filter process is initially centered on pulse 24 (using pulses 21 to 27) and ends with pulse 348 (using pulses 345 to 351).

The identification of peaks in the filtered scan is restricted to pulses between pulse count 24 and 348. Simply stated, a series of pulses constitutes a valid peak if the pulse magnitudes are increasing for at least four consecutive pulses and decreasing for the two succeeding pulses. When a peak is identified, the pulse count and the magnitude at the point of slope reversal (indicating the summit of the peak) are stored for later analysis. After the fully filtered scan has been examined, the magnitudes of the stored peaks are ranked in descending order. The program then stores the peak count with largest magnitude for latter analysis. In its R&D

capacity, this automated basic processing system served the needs of a follow-on study, which used the largest and second highest peak in the evaluation of automated cloud observation systems.[5] Therefore, the second largest peak was also stored. If no peaks were identified during the scan, the program stored 1000 for both the maximum and secondary maximum peaks. However if a maximum peak exists and a secondary peak does not, 1000 is stored only for the secondary maximum.

The determination of the objective cloud base height is initiated after five scans have been obtained for a lamp. Then after each successive scan an objective cloud base height is calculated for each lamp using the maximum peak pulse count determined for each of the most recent five scans. Recall that five scans of one lamp span 1 min in total time. Initially, the five peaks are compared and if two or more of them have the same pulse count, that count is selected as the cloud base height. If two values have multiple occurrences, the lowest is selected. For example, if peak pulse count values of 140, 120, 120, 100, and 100 are reported, 100 is selected as the cloud base height. If all five values are different, a second check is made to determine if two or more pulse counts are within 4 counts of each other. If they are, the lowest pulse count is stored as the cloud base height. For example, if values of 92, 54, 96, 97, and 52 are reported, 52 is stored. If the first two tests fail, a final attempt is made. If but one of the five values is within 6 pulse counts of the value identified in the previous cloud base height determination for the lamp under consideration, then the lower of these two values is retained. If all three tests fail then the cloud base height is reported as none by storing 1000. At this point in the program, a check is made to determine if the one-scan or the five-scan mode has been selected. In the one-scan mode, the five-scan cloud base height determination is made but not used. This permits a rapid update when the five-scan mode is reselected. The pulse height count, representing the largest return received in the current scan, is converted to a height in feet using Appendix C. The numeric characters for the height, the stored cursor position, and the stored display array are then sent to update the dot matrix display. If the magnitudes of the data points were zero, "NO PEAK" would be displayed for the height and the cursor would be suppressed. In the five-scan mode, the most recent stored data array, the stored cursor position, and the numeric characters for the cloud base height are displayed. Again, the CBH (in feet) is determined using Appendix C. If a cloud base height was determined to be none or produced by sunglint, "NO CBH" is displayed.

After the data for a given lamp have been analyzed and displayed, the program completes the collection of offset data and waits for the other lamp's zero degree

---

5. Geisler, E.B. and Chisholm, D.A. (1980) An Automatic Cloud Observation System (ACOS), AFGL-TR-81-0002, AD A094 330.

switch closure. When the zero degree switch closure occurs, cloud base data collection proceeds using the interrupt mode for the new lamp. An offset correction for the previous lamp is calculated and stored for the next cycle while the basic processing described above continues for the new lamp.

The third and final mode of operation is the manual mode. While this mode continues to allow data to be collected, it inhibits the display from updating, thereby allowing the operator to evaluate the current display at length. The cursor positioner, a front panel switch, is enabled so that the operator can move the cursor to the left or to the right to investigate secondary maxima or other interesting features in the trace. When the positioner is toggled and not held, the cursor moves only one column to allow exact positioning. However, if the switch is held in the toggled position, the cursor moves very rapidly to permit longer moves. In either the toggled or the toggled and hold operation, a conversion of the cursor position to the equivalent height in feet is accomplished and automatically updated in the display's upper right corner.

5. FUTURE CONSIDERATIONS

In the studies conducted thus far, an effective method to process and report an obscuration has not been found. This is largely due to the fact that the RBC's response to an obscuration varies tremendously depending on the weather element causing it and its intensity and thickness. Procedures tested thus far, in the demonstration of the Modular Automated Weather System at Scott AFB, IL[6] and more recently at AFGL and Otis AFB, have not yielded a fully adequate solution. Data being gathered from a three RBC network at AFGL's Weather Test Facility on Otis AFB, MA are being used to try to resolve this difficult automation task. When developed, obscuration procedures could be incorporated into the processing program.

In a related study,[5] hierarchical clustering techniques for automated cloud observations have been evaluated using data gathered by the three RBC network at the Weather Test Facility at Otis AFB. This clustering technique, applied to the latest 30 min of cloud data provided by our RBC processing procedures, yielded reliable and representative observation of low cloud height, low cloud amount, and ceiling. It also found that specification accuracy is not measurably improved by using basic data from a second or third nearby RBC and that the use of secondary peak information does not add to improved specification. It is our contention,

---

6. Chisholm, D.A., Lynch, R.H., Weyman, J.C., and Geisler, E.B. (1980) A Demonstration Test of the Modular Automated Weather System (MAWS), AFGL-TR-80-0087, AD A087 070.

however, that if secondary peaks survive the filters and tests applied to them in the program, they could be valuable information in support of aircraft operations. Expanding the processing and display capability to include these added pieces of information (low cloud height, amount and ceiling based on updated 30 min data sets and the secondary peaks) would require added display space beyond that presently included. An alternative solution would allow the operator to display low cloud observations and related data using a front panel switch which could direct the computer to temporarily hold RBC scans and display alphanumeric data instead.

Another change would be required in order to implement these modifications throughout the Air Weather Service inventory. Procedures are needed to account for differences in the RBC baseline (distance from projector to detector) and line frequency which can be 50 cps (cycles per second) at some overseas bases compared to 60 cps in the U.S. These variations can be accounted for through internal switches which can be set during installation consistent with local conditions.

Finally, a test routine, which would be implemented at the option of the observer-operator, could be included in the system. Its purpose would be to assist in the diagnosis of suspected system malfunctions. When the program appears to be functioning improperly, a known data set would be generated by an internal signal simulator. Comparison of anticipated outcomes with actual outcomes of the several processing steps would aid the operator in isolating system malfunctions (if any). Work is presently underway to incorporate the test routine and the inventory-wide option capability into the display system.

6. SUMMARY AND CONCLUSIONS

Although the RBC is an aging sensor, with proper maintenance and handling it should be useful to the AWS for several more years. Further, the cost to completely replace the RBCs in the Air Force inventory would be very high. This study has demonstrated the advantages of using a low cost microprocessor-based automated display system to replace the display portion of the current system. Although the prototype display system can continue to undergo modifications and improvements, it goes a long way towards overcoming the operational and maintenance problems encountered with the current CRT display.

In addition to solving the maintenance problems, the display/processor provides an added dimension that was not previously available to the operator. It provides a reliable, repeatable scan which minimizes the subjectivity of the operator and the misalignment of certain components. In addition, the system displays the exact height of the maximum return for each scan to an accuracy never obtainable before. Fine scale variations can also be seen. The system provides the operator with options to select one of three modes to process and display the RBC data. Two modes are fully automated: a one-scan mode which operates only on the latest scan, and a five-scan mode which objectively determines an updated 1 min cloud base height. The third, a manual-interactive mode, allows the observer to hold the current scan and fully investigate, by means of a moving cursor, primary peaks, secondary peaks, or other areas of interest. This mode permits the observer to extract in detail all the information a scan has to offer.

In November 1980, a prototype RBC display system was installed in the Base Weather Station (BWS) at Scott AFB, IL for a MAC Service Test. It was interfaced to the operational RBC sensors and was located in close proximity to the operational CRT display. Operating procedures and evaluation criteria were briefed to BWS, Hq AWS and AFCC personnel. From our experience with the system and from responses received from Scott personnel, the automated display system has proven to be very beneficial. The display is more readable, provides more accurate heights, and permits better interpretation of the data obtained. In its present form, or more so with the improvements suggested above, this display system can significantly enhance the utility of the operational RBC and is a step towards more accurate and reliable observations.

References

1. Air Force Communications Service (1977) Automated Weather Distribution System (AWDS) Required Operational Capability (AFSC ROC 601-77).

2. Burroughs Bulletin No. 3511A, Burroughs Corporation, OEM Division, P.O. Box 1226, Plainfield, New Jersey 07061.

3. Moroz, E. Y., and Travers, G.A. (1975) Measurement of Cloud Height, AFCRL-TR-75-0306, AD A015 737.

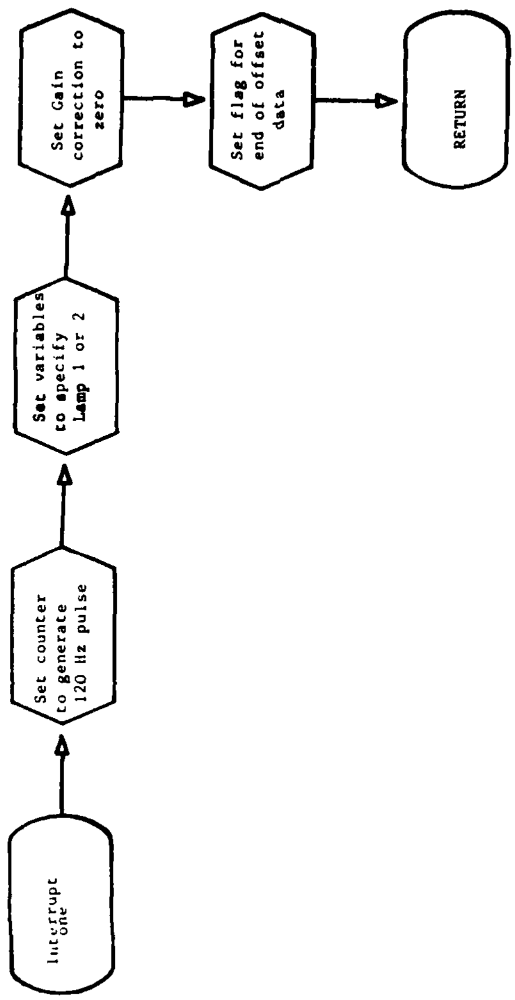
4. Federal Meteorological Handbook No. 1, Surface Observations, Second Edition (1979), Washington, DC, pp A5-5 to A5-29.

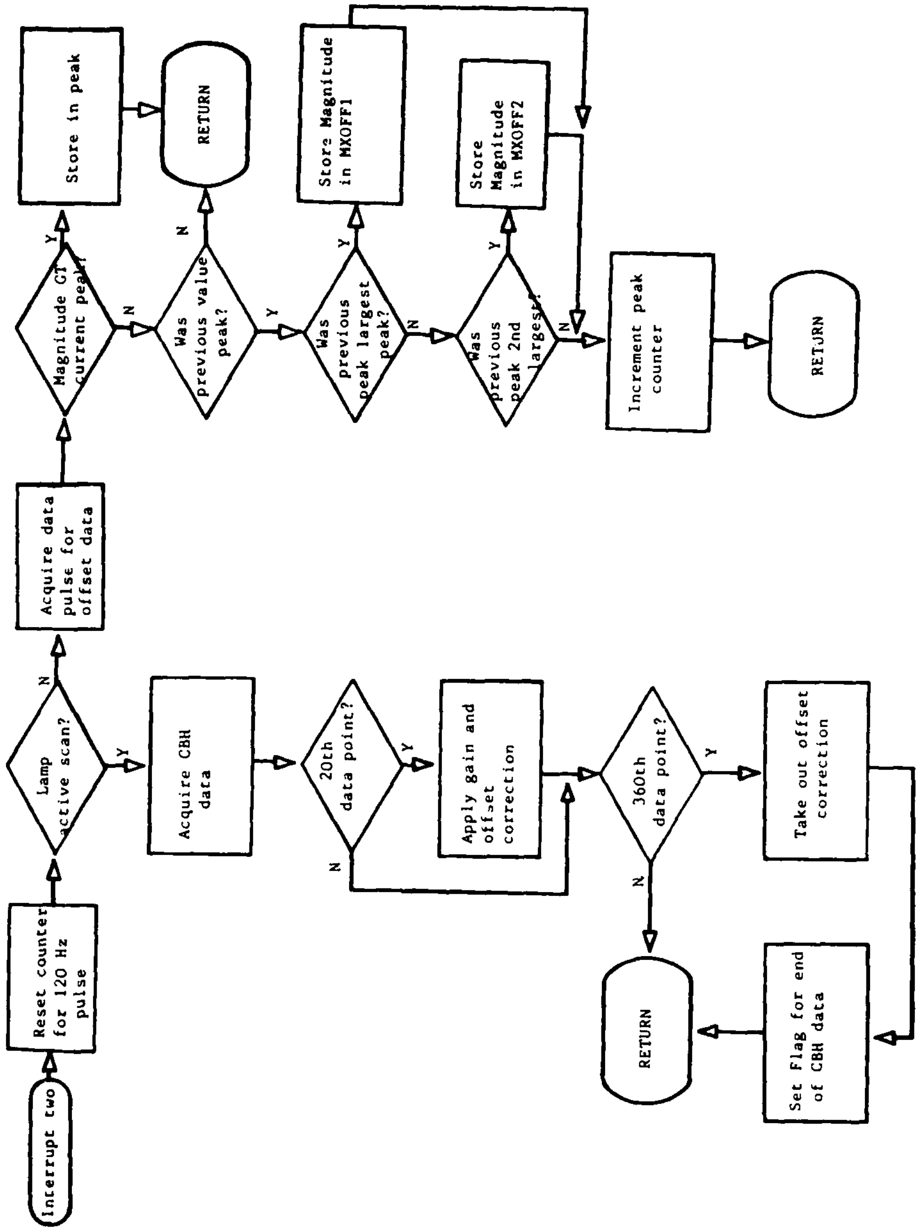
5. Geisler, E.B. and Chisholm, D.A. (1980) An Automated Cloud Observation System (ACOS), AFGL-TR-81-0002, AD A094 330.

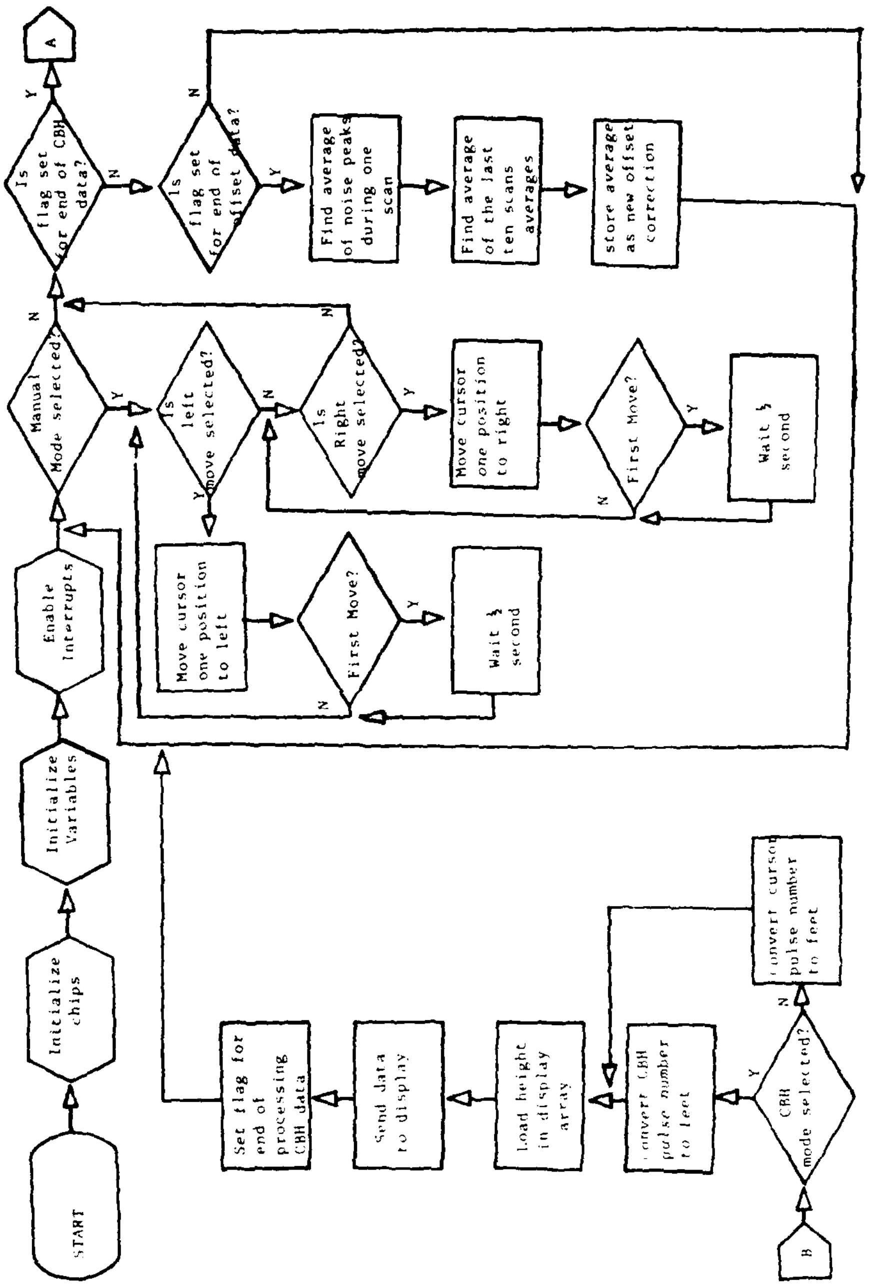
6. Chisholm, D.A., Lynch, R.H., Weyman, J.C., and Geisler, E.B. (1980) A Demonstration Test of the Modular Automated Weather System (MAWS), AFGL-TR-80-0087, AD A087 070.

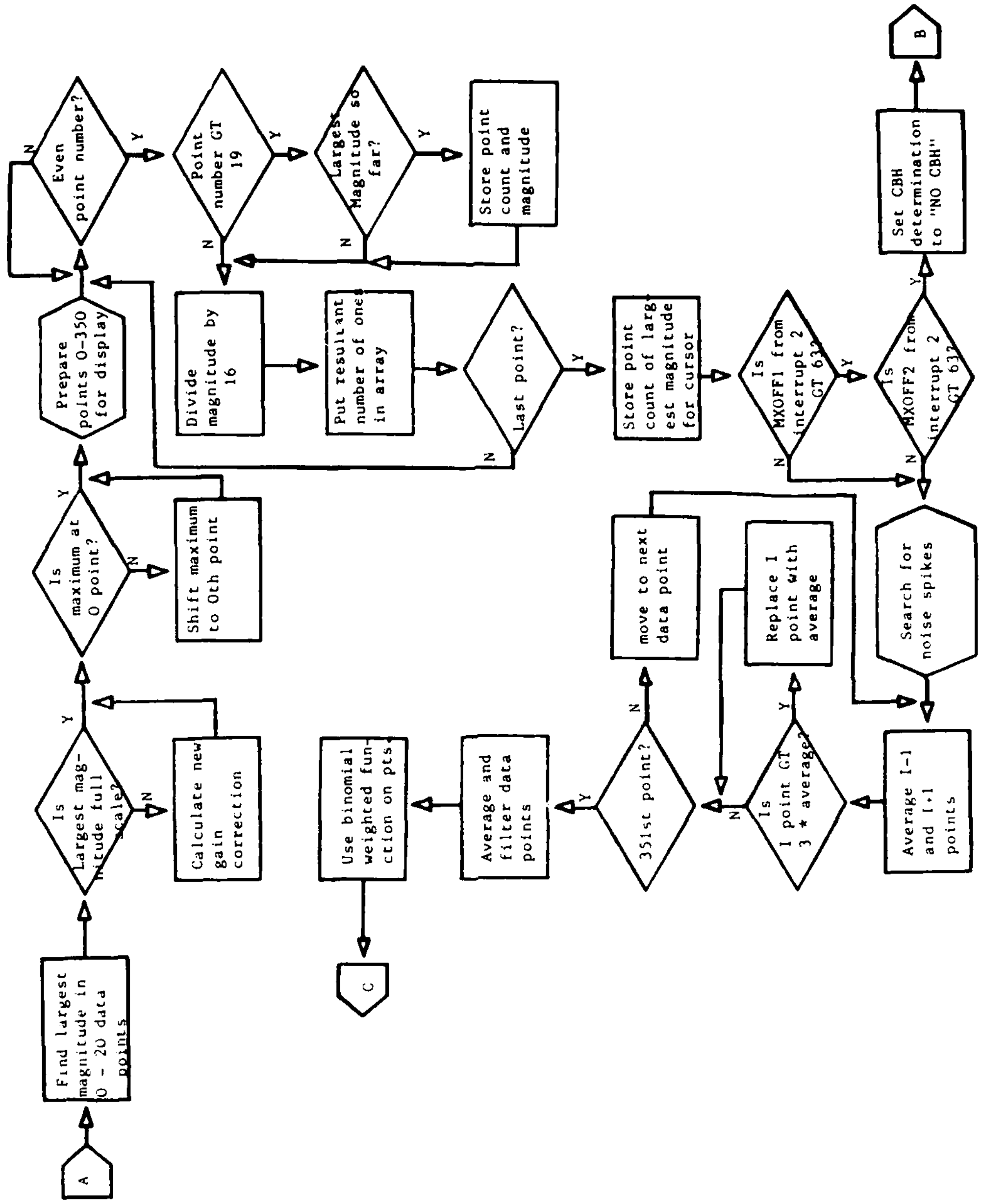
Appendix A

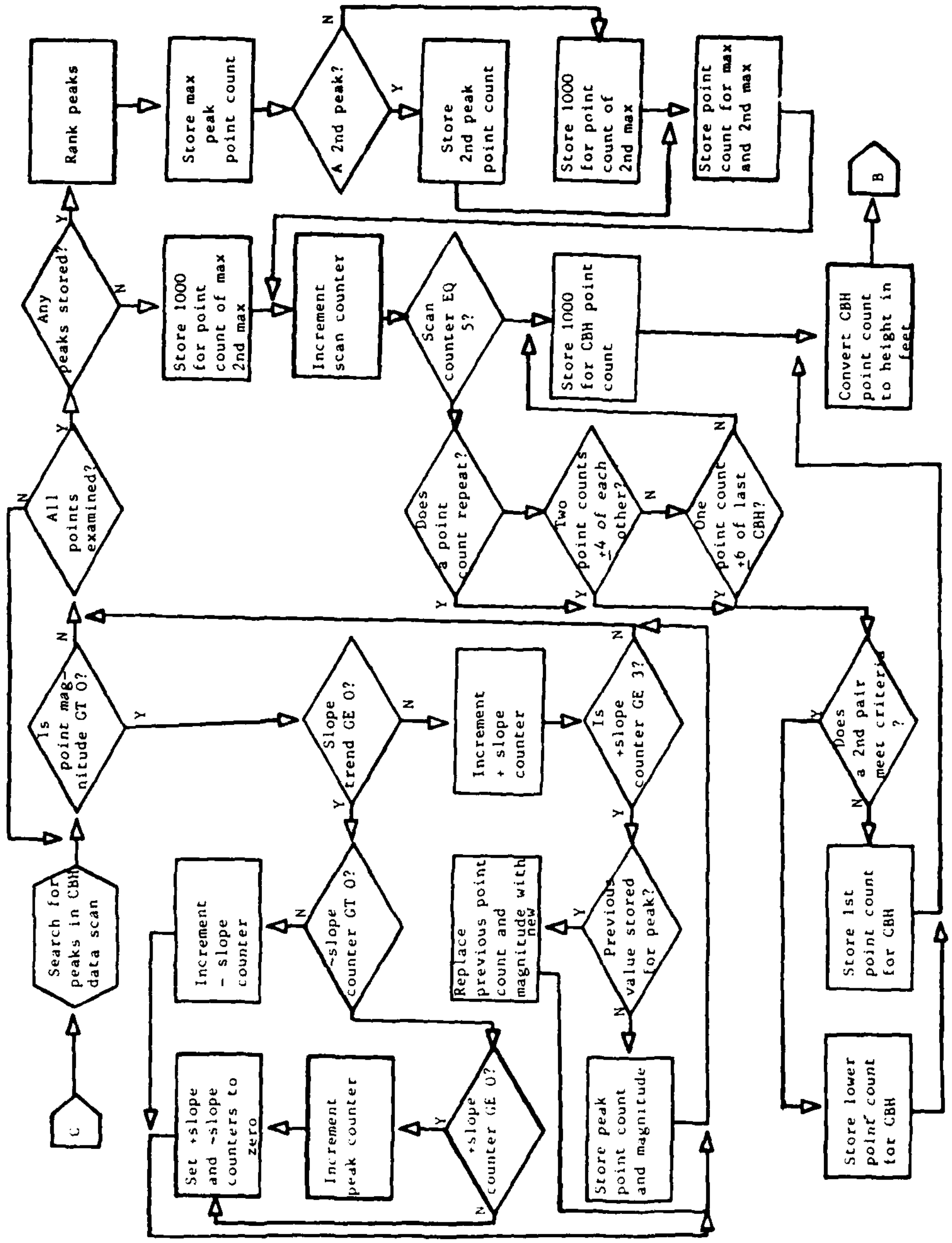
Flow Chart of Processing Program


Interrupt one
Set counter to generate 120 Hz pulse
Set variables to specify Lamp 1 or 2
Set Gain correction to zero
Set flag for end of offset data
RETURN


Interrupt two
Reset counter for 120 Hz pulse
Lamp active scan?
N
Y
Acquire data pulse for offset data
Magnitude GT current peak?
Y
N
Store in peak
RETURN
Was previous value peak?
N
Y
Was previous peak largest peak?
Y
N
Store Magnitude in MXOFF1
Was previous peak 2nd largest?
Y
N
Store Magnitude in MXOFF2
Increment peak counter
RETURN
Acquire CBH data
20th data point?
Y
N
Apply gain and offset correction
360th data point?
Y
N
RETURN
Take out offset correction
Set Flag for end of CBH data
RETURN


START
Initialize chips
Initialize Variables
Enable Interrupts
Manual Mode selected?
Is flag set for end of CBH data?
A
Is flag set for end of offset data?
Find average of noise peaks during one scan
Find average of the last ten scans averages
Store average as new offset correction
Is left move selected?
Is Right move selected?
Move cursor one position to right
First Move?
Wait ½ second
Move cursor one position to left
First Move?
Wait ½ second
B
CBH mode selected?
Convert cursor pulse number to feet
Convert CBH pulse number to feet
Load height in display array
Send data to display
Set flag for end of processing CBH data
Y
N


A
Find largest magnitude in 0 - 20 data points
Is Largest magnitude full scale?
Calculate new gain correction
Is maximum at 0 point?
Shift maximum to 0th point
Prepare points 0-350 for display
Even point number?
Point number GT 19
Largest Magnitude so far?
Store point count and magnitude
Divide magnitude by 16
Put resultant number of ones in array
Last point?
Store point count of largest magnitude for cursor
Is MXOFF1 from interrupt 2 GT 63?
Is MXOFF2 from interrupt 2 GT 63?
Set CBH determination to "NO CBH"
B
Search for noise spikes
Average I-1 and I+1 points
Is I point GT 3 * average?
Replace I point with average
351st point?
move to next data point
Average and filter data points
Use binomial weighted function on pts.
C
Y
N


C
Search for peaks in CBH data scan
Is point magnitude GT 0?
All points examined?
Any peaks stored?
Rank peaks
Store max peak point count
A 2nd peak?
Store 2nd peak point count
Store 1000 for point count of 2nd max
Store point count for max and 2nd max
Store 1000 for point count of max 2nd max
Increment scan counter
Scan counter EQ 5?
Store 1000 for CBH point count
Convert CBH point count to height in feet
B
Does a point count repeat?
Two point counts ±4 of each other?
One point count ±6 of last CBH?
Does a 2nd pair meet criteria?
Store 1st point count for CBH
Store lower point count for CBH
Slope trend GE 0?
Increment + slope counter
Is +slope counter GE 3?
Previous value stored for peak?
Replace previous point count and magnitude with new
Store peak point count and magnitude
-slope counter GT 0?
Increment - slope counter
+slope counter GE 0?
Increment peak counter
Set +slope and -slope counters to zero
Y
N

Appendix B

PLM-80 Software of Processing Program

```
PL/M-80 COMPILER    RBC DISPLAY - MAIN PROGRAM

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE RBCDISPLAYPROGRAM
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:RBCDIS.SAV TITLE('RBC DISPLAY - MAIN PROGRAM') NOOBJECT


                 $PRINT(:LP:)  DEBUG  INTVECTOR(4,40H)

   1             RBC$DISPLAY$PROGRAM:  DO;

   2    1        DECLARE  DCL LITERALLY  'DECLARE';

   3    1        DCL (CHECK,CNT,COFF,CURSOR,DONE,GETOFF,J,K1,K2,K3,K4,K5,K6,K7,LEFT) BYTE PUBLIC;
   4    1        DCL (MANUAL,MAX,MAXI,MAXN,MXOFF1,MXOFF2,NO,NO2,OFFS,RFLAG) BYTE PUBLIC;
   5    1        DCL (RIGHT,SCAN,X) BYTE PUBLIC;
   6    1        DCL (CTR,I,IC,K,POFF) ADDRESS PUBLIC;

   7    1        DCL A(361) BYTE PUBLIC, BEIL(2) ADDRESS PUBLIC, C(20) BYTE PUBLIC;
   8    1        DCL CEIL(2) ADDRESS PUBLIC, CLOUD(2) ADDRESS PUBLIC, CLOUDY(22) ADDRESS PUBLIC;
   9    1        DCL DIS(200) ADDRESS PUBLIC, GAIN(2) BYTE PUBLIC, GRAND(2) BYTE PUBLIC;
  10    1        DCL HEIT(20) ADDRESS PUBLIC, OCT(2) BYTE PUBLIC, OFFSET(2) BYTE PUBLIC;
  11    1        DCL PEAK(80) BYTE PUBLIC, PEAKO(200) BYTE PUBLIC, PULSE(80) ADDRESS PUBLIC;
  12    1        DCL RCT(2) BYTE PUBLIC;

  13    1        DCL HEAD(16) ADDRESS PUBLIC DATA(
                 01H,03H,07H,0FH,1FH,3FH,7FH,0FFH,1FFH,3FFH,7FFH,0FFFH,1FFFH,3FFFH,7FFFH,0FFFFH);

  14    1        CHECK1: PROCEDURE(NO2) EXTERNAL;
  15    2        DCL NO2 BYTE;
  16    2        END CHECK1;

  17    1        DATA$CHECK: PROCEDURE(P,Y,Z) EXTERNAL;
  18    2        DCL (Y,Z) BYTE, P ADDRESS;
  19    2        END DATA$CHECK;

  20    1        DETOFF: PROCEDURE EXTERNAL;
  21    2        END DETOFF;

  22    1        FILL: PROCEDURE EXTERNAL;
  23    2        END FILL;

  24    1        GAINER: PROCEDURE(X,Y) EXTERNAL;
  25    2        DCL (X,Y) BYTE;
  26    2        END GAINER;

  27    1        INTONE: PROCEDURE EXTERNAL;
  28    2        END INTONE;

  29    1        INTTWO: PROCEDURE EXTERNAL;
  30    2        END INTTWO;

  31    1        MTIME: PROCEDURE(X) EXTERNAL;
  32    2        DCL X BYTE;
  33    2        END MTIME;

  34    1        OFFCAL: PROCEDURE EXTERNAL;
  35    2        END OFFCAL;
```

```
 36   1        RESET: PROCEDURE EXTERNAL;
 37   2        END RESET;

 38   1        SCAN5: PROCEDURE(NO2) EXTERNAL;
 39   2        DCL NO2 BYTE;
 40   2        END SCAN5;

 41   1        VARY:  PROCEDURE EXTERNAL;
 42   2        END VARY;

 43   1        INTER1: PROCEDURE INTERRUPT 1;
 44   2        CALL INTONE;
 45   2        MEMORY(0000H) = 20H;
 46   2        END INTER1;

 47   1        INTER2: PROCEDURE INTERRUPT 2;
 48   2        CALL INTTWO;
 49   2        MEMORY(0000H) = 20H;
 50   2        END INTER2;

 51   1        BEGIN:  DISABLE;
 52   1        MEMORY(8403H) = 30H;  /*8253 COUNTER 0 LD/RD*/
 53   1        MEMORY(8400H) = 0FFH; /*8253 LS BYTE LOAD*/
 54   1        MEMORY(8400H) = 0FFH; /*8253 MS BYTE LOAD*/
 55   1        MEMORY(8C00H) = 0E7H; /*PROGRAM BAUD RATE 1200*/
 56   1        MEMORY(0000H) = 56H;  /*8259 ICW1  ADDRESS START 0040H,EDGE TIGGERED,INTERVAL 4,SINGLE, NO ICW4*/
 57   1        MEMORY(0001H) = 00H;  /*8259 ICW2  TOP BYTE OF ADDRESS*/
 58   1        MEMORY(0001H) = 0FFH; /*MASK FOR 8259*/
 59   1        MEMORY(0801H) = 7AH;  /*8251 MODE WORD  CLOCK/16, 7 BIT, PARITY DISABLED, EVEN PARITY, ONE STOP BIT*/
 60   1        MEMORY(0801H) = 35H;  /*8251 COMMAND WORD: XMIT ENABLE,RECVR ENABLE,RESET ERR FLAGS,REQUEST TO SEND*/
 61   1        I = MEMORY(0800H);    /*CLEAR RECEIVER BUFFER*/

 62   1        MEMORY(6603H) = 9AH;  /*FORMAT 8255 FOR RBC*/
 63   1        MEMORY(6600H) - 00H;  /*SET BIAS LEVEL 0 OF 16*/
 64   1        MEMORY(6602H) = 00H;  /*SET GAIN OF 1.00*/

               /*INITIALIZE VARIABLES FOR PROPER STARTING CONDITIONS*/

 65   1        CHECK = 0AAH;
 66   1        GAIN(0),GAIN(1) = 00H;
 67   1        OFFSET(0),OFFSET(1) = 02H;
 68   1        BEIL(0),BEIL(1),CEIL(0),CEIL(1) = 1000;
 69   1        RCT(0),RCT(1),OCT(0),OCT(1) = 0;
 70   1        GRAND(0),GRAND(1),CLOUD(0),CLOUD(1) = 0;
 71   1        K1,K2,K3,K4,K5,K6,K7 = 0;
 72   1        CNT,COFF,IC,MXOFF1,MXOFF2 = 0;
 73   1        DONE,GETOFF,POFF = 1;
 74   1        RFLAG = 4;
 75   1        IC = 362;
 76   1        MEMORY(0001H) = 0F9H;              /*SET MASK FOR 8259:ALLOW INTER 1 AND 2*/
 77   1        ENABLE;
 78   1        LOOP: MEMORY(8000H) = 00H;          /*SYSTEM RUN LIGHT*/
 79   1        MANUAL = MEMORY(6601H) AND 02H;     /*WHEN THE SYSTEM IS PLACED IN MANUAL MODE BIT 1 OF THIS
                                                    LOCATION BECOMES 1*/
```

```
 80  1   IF MANUAL = 02H THEN DO;             /*DO IF IN MANUAL MODE*/
 82  2   DO WHILE MANUAL = 02H;               /*REMAIN IN LOOP UNTIL TAKEN OUT OF MANUAL MODE*/
 83  3   MANUAL = MEMORY(6601H) AND 02H;
 84  3   LEFT = MEMORY(6601H) AND 04H;        /*WHEN IN MANUAL MODE AND TOGGLE SWITCH IS PLACED IN LEFT
                                              CURSOR THEN BIT TWO WILL BECOME TRUE*/
 85  3   RIGHT = MEMORY(6601H) AND 08H;       /*WHEN IN MANUAL MODE AND THE TOGGLE SWITCH
                                              IS PLACED IN RIGHT CURSOR THEN BIT THREE WILL BECOME TRUE*/
 86  3   CTR = 0;                             /*COUNTER TO MAKE FIRST MOVE SLOW AND THEN SPEED UP*/
 87  3   DO WHILE LEFT = 04H;                 /*CONTINUE TO DO UNTIL FALSE*/
 88  4   IF CURSOR > 0 THEN DO;               /*CONTINUE TO MOVE LEFT UNTIL AT 0 DEGREES*/
 90  5   DIS(CURSOR) = NOT DIS(CURSOR);       /*CORRECT LAST CURSOR POSITION TO REGULAR VIDEO*/
 91  5   IF CURSOR = 192 THEN CURSOR = 175;
 93  5   ELSE CURSOR = CURSOR - 1;                 /*CHANGE CURSOR POSITION TO THE LEFT*/
 94  5   IF NO = 1 THEN DO;                   /*NO PEAK OR NO CBH WAS LAST THING PRINTED, AND
                                              NOW MUST CLEAR THAT AREA IN MEMORY*/
 96  6   NO = 0;
 97  6   DO I = 151 TO 165;
 98  7   DIS(I) = DIS(I) AND 00FFH;
 99  7   END;  END;
101  5   CALL FILL;                           /*FILL ARRAY AND SEND TO DISPLAY*/
102  5   CALL RESET;                          /*CHECK ON DONE BECAUSE WE ARE NOT GOING TO PROCESS DATA*/
103  5   END;
104  4   CALL VARY;                           /*CHECK IF A SLOW MOVE IS REQUESTED OR FAST ON*/
105  4   LEFT = MEMORY(6601H) AND 04H;        /*CHECK IF TOGGLE IS STILL ENGAGED*/
106  4   END;
107  3   DO WHILE RIGHT = 08H;                /*DO UNTIL TOGGLE IS DISENGAGED*/
108  4   IF CURSOR < 175 THEN DO;             /*ONLY RIGHT AS FAR AS DATA IS SHOWN*/
110  5   DIS(CURSOR) = NOT DIS(CURSOR);       /*CORRECT POSITION OF LAST CURSOR SO IT BECOMES REGULAR VIDEO*/
111  5   CURSOR = CURSOR + 1;                 /*INCREASE CURSOR COUNT BY ONE*/
112  5   CALL FILL;                           /*FILL NEW ARRAY AND DISPLAY*/
113  5   CALL RESET;                          /*CHECK ON STATUS OF DONE; WILL NOT PROCESS DATA*/
114  5   END;
115  4   CALL VARY;                           /*FAST OR SLOW MOVE*/
116  4   RIGHT = MEMORY(6601H) AND 08H;       /*CHECK IF TOGGLE IS STILL ENGAGED*/
117  4   END;  END;  END;
120  1   IF DONE = 0 AND (RFLAG = 2 OR RFLAG = 4) THEN DO;   /*CBH DATA HAS ALL BEEN RECEIVED AND MUST BE
                                                PROCESSED*/
122  2   RCT(K3) = RCT(K3) + 1;                 /*RUNNING TOTAL OF THE NUMBER OF SCANS THAT HAVE BEEN
                                                PROCESSED TO NOW; MAX OF 5*/
123  2   GRAND(K3),MAX = 0;                     /*GRAND WILL BE USED TO STORE THE INTENSITY OF LARGEST PEAK
                                                FOUND IN THE FIRST 4.75 DEGREES*/
                                                /*MAX WILL BE USED TO STORE THE LOCATION OF THE LARGEST PEAK
                                                IN THE FIRST 4.75 DEGREES*/
124  2   DO I = 0 TO 19;                        /*DO LOOP TO FIND THE MAX INTENSITY IN THE FIRST 4.75 DEGREES*/
125  3   IF A(I) > GRAND(K3) THEN DO;           /*LARGER MAGNITUDE FOUND IN THE SCAN*/
127  4   GRAND(K3) = A(I);                      /*STORE MAGNITUDE*/
128  4   MAX = I;                               /*STORE LOCATION*/
129  4   END;  END;
131  2   CALL GAINER(GRAND(K3),K3);             /*CALCULATE A NEW GAIN CORRECTION BASED ON THE VALUE OF GRAND*/
132  2   IF MAX > 0 AND MAX < 9 THEN DO;        /*IF MAX INTENSITY WAS NOT AT ZERO DEGREES SHIFTED IT;
                                                THE MOST IT CAN BE SHIFTED IS 8 POINTS (2 DEGREES)*/
134  3   DO I = 0 TO (360 - MAX);               /*DO LOOP TO SHIFT THE DATA POINTS BECAUSE THE MAX INTENSITY
                                                WAS NOT AT ZERO DEGREES*/
135  4   A(I) = A(I+MAX);
136  4   END;  END;
138  2   MAXI = 0;                              /*MAXI IS THE MAXIMUM INTENSITY OF THE CBH DATA COLLECTED
                                                DURING THE LAST SCAN; USED WITH CURSOR PLACEMENT*/
```

```
139  2     MAXN = 192;                          /*MAXN IS THE MAXIMUM NUMBER WHICH IS THE COUNT OF THE MAX
                                                 PEAK IN THE SACN*/
140  2     DO I = 0 TO 350 BY 2;                /*DO LOOP TO GO THROUGH THE CBH DATA TO FIND POSITION OF
                                                 MAX PEAK AND VALUE THERE*/
141  3     K = I / 2;                           /*K IS A DUMMY VARIABLE FOR PLACEMENT OF CURSOR AND MAXN
                                                 SINCE THE DISPLAY ONLY SHOWS EVERY OTHER DATA POINT*/
142  3     IF A(I) = 0 THEN DIS(K) = 0;         /*IF CBH DATA POINT IS ZERO PUT ZERO IN DISPLAY ARRAY*/
144  3     ELSE DO;                             /*ALL OTHER NON ZERO CBH DATA POINTS*/
145  4     IF I > 19 THEN DO;                   /*DO NOT WANT A PEAK IN FIRST FIVE DEGREES BECAUSE THAT IS
                                                 WHERE THE START BANG IS*/
147  5     IF A(I) > MAXI THEN DO;              /*CURRENT CBH DATA POINT IS LARGER IN MAGNITUDE THAN PREVIOUS
                                                 MAX FOUND*/
149  6     MAXI = A(I);                         /*STORE MAGNITUDE OF MAX FOUND*/
150  6     MAXN = K;                            /*STORE COUNT NUMBER OF MAX FOUND*/
151  6     END; END;
153  4     DIS(K) = HEAD(A(I)/16);              /*RANGE OF THE DIGITAL VALUES IS 0 -255 HOWEVER ONLY 0 - 15
                                                 CAN BE SHOWN ON DISPLAY SO DIVIDE BY 16*/
154  4     END;  END;
156  2     CURSOR = MAXN;                       /*SET CURSOR TO COUNT OF THE MAX MAGNITUDE FOUND*/

           /*BASED ON PAST EXPERIENCE ON A SUNNY DAY THE RBC SIGNAL IS VERY NOISY AND ALOT
           OF FALSE RETURNS ARE REPORTED AS PEAKS. TO ELIMINATE THOSE RETURNS A CHECK OF THE
           MAGNITUDE OF THE TWO LARGEST PEAKS DURING THE OFF SCAN. IF THESE TWO ARE FOUND
           TO BE OVER 1/4 SCALE THEN A SUNNY DAY IS ASSUMED AND THE MAX AND 2ND MAX FOR
           THE PREVIOUS FIVE SCANS ARE SET TO A NO PEAK OR A NO CBH CONDITION*/

157  2     IF MXOFF1 > 63 THEN DO;            /*FIRST MAX GREATER THAN A 1/4 SCALE*/
159  3     IF MXOFF2 > 63 THEN DO;            /*SECOND MAX GREATER THAN A 1/4 SCALE*/
161  4     IF BEIL(K3) > 348 THEN DO;         /*PREVIOUS CBH WAS "NO CBH"*/
163  5     K4 = K3 * 10;                      /*K4 WILL FIND THE CORRECT PLACE IN HEIT*/
164  5     RCT(K3) = 5;                       /*MAKE SURE A NO CBH IS REPORTED AS A 5 SCAN VALUE*/
165  5     DO I = 0 TO 9;                     /*DO LOOP TO PLACE THE VALUE FOR NO PEAKS IN HEIT*/
166  6     HEIT(K4+I) = 1000;
167  6     END;  END;  END;  END;
171  2     CALL CHECK1(K3);                   /*DETERMINE THE MAX AND 2ND MAX PEAK FOR THE CURRENT SCAN
                                               AFTER FILTERING AND AVERAGING*/
172  2     IF RCT(K3) = 5 THEN CALL SCAN5(K3);  /*FINDS A 5 SCAN REPRESENTATIVE VALUE*/
174  2     CALL FILL;
175  2     CALL RESET;

176  2     END;
177  1     IF GETOFF = 0 THEN DO;               /*THE OFFSCAN NOISE DATA FOR OFFSET DETERMINATION HAS
                                                 ALL BEEN COLLECTED*/
179  2     POFF,K1 = 0;                         /*POFF WILL BE THE SUM OF ALL THE NOISE PEAKS COLLECTED
                                                 DURING THE OFFSCAN AND K1 WILL BE HOW MANY PEAKS
                                                 WENT INTO THE TOTAL*/
180  2     DO K2 = 0 TO COFF;                   /*DO LOOP TO FIND TOTAL OF NOISE PEAKS*/
181  3     IF PEAKO(K2) > 0 THEN DO;
183  4     POFF = POFF + PEAKO(K2);             /*RUNNING TOTAL IN POFF*/
184  4     K1 = K1 + 1;                         /*RUNNING TOTAL OF NUMBER OF PEAKS GOING INTO TOTAL*/
185  4     END;  END;
187  2     IF K1 > 0 THEN OFFS = POFF/K1;       /*IF ANY NOISE PEAKS FIND THE AVERAGE OF ALL */
189  2     ELSE OFFS = 0;                       /*IF NONE SET AVERAGE TO ZERO*/
190  2     CALL DETOFF;                         /*DETERMINE NEXT OFFSET CORRECTION FOR LAST LAMP*/
191  2     GETOFF = 1;                          /*OFFSET NOISE DATA HAS BEEN PROCESSED*/
192  2     END;
193  1     GO TO LOOP;
```

```
PL/M-80 COMPILER    RBC DISPLAY - MAIN PROGRAM

 194   1      END RBC$DISPLAY$PROGRAM;


MODULE INFORMATION:

     CODE AREA SIZE     = 04B6H   1206D
     VARIABLE AREA SIZE = 0555H   1365D
     MAXIMUM STACK SIZE = 000AH     10D
     223 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - ASCII

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE ASCIIMOD
OBJECT MODULE PLACED IN :F1:ASCII.OBJ
COMPILER INVOKED BY:  PLM80 :F1:ASCII.SRC TITLE('RBC DISPLAY - ASCII')


              $PRINT(:LP:)  DEBUG

 1            ASCII$MOD:  DO;

              /*THIS PROCEDURE TAKES THE POSITION OF THE PEAK AS SHOWN BY THE CURSOR IN
              THE ONE SCAN MODE OR THE POSITION OF THE DETERMINED CBH BOTH OF WHICH ARE
              IN 1/4 DEGREE INCREMENTS AND CONVERTS IT TO A HEIGHT IN FEET. THEN THE FEET
              HEIGHT IS CONVERTED TO A DOT PATTERN ON THE DISPLAY WHICH WILL PRINT OUT
              THE APPROPRIATE NUMBER. */

 2    1       DECLARE DCL LITERALLY 'DECLARE';

 3    1       DCL NO BYTE EXTERNAL;
 4    1       DCL DIS(200) ADDRESS EXTERNAL;
 5    1       DCL HEAD(118) BYTE DATA(3EH,41H,41H,41H,3EH,01H,21H,7FH,01H,01H,
              21H,43H,45H,49H,31H,22H,41H,49H,49H,36H,0CH,14H,24H,7FH,04H,72H,51H,51H,51H,4EH,
              3EH,49H,49H,49H,06H,41H,42H,44H,48H,70H,36H,49H,49H,49H,36H,30H,49H,4AH,4CH,38H,
              7FH,20H,1CH,02H,7FH,00H,3EH,41H,41H,41H,3EH,00H,00H,00H,00H,00H,00H,00H,0FFH,0FFH,
              7FH,48H,48H,48H,30H,00H,7FH,49H,49H,49H,41H,00H,1FH,24H,44H,24H,1FH,00H,7FH,08H,
              14H,22H,41H,0FFH,0FFH,3EH,41H,41H,41H,22H,00H,7FH,49H,49H,49H,37H,00H,7FH,08H,08H,
              08H,7FH,00H,00H,00H,00H,00H,00H);

              /*HEAD IS A FIVE BY SEVEN MATRIX WHICH WRITES OUT ON THE DISPLAY

              0 1 2   4 5 6 7 8 9 N O   P E A K   C B H
              DUMMY CHARACTERS OF 0FFH ARE ADDED AFTER NO  AND PEAK TO MAKE THE COUNT BETTER
              WHEN USED IN A DO LOOP*/

 6    1       HEIGHT: PROCEDURE(Y) ADDRESS EXTERNAL;
 7    2       DCL Y ADDRESS;
 8    2       END HEIGHT;

 9    1       PUT: PROCEDURE(X,Y,Z);
10    2       DCL (I,K,X,Y,Z) BYTE, J ADDRESS;
11    2       K = 0;                              /*K IS A DUMMY VARIABLE TO COUNT THRU HEAD THE CORRECT NUMBER*/
12    2       Z = Z * 5;                          /*Z IS THE NUMBER TO BE CONVERTED INTO A 5 X 7 NUMBER; TO FIND
                                                  CORRECT PLACE MULTIPLY BY FIVE. ALSO AT THE END IS THE WORDS NO,
                                                  PEAK, AND CBH WITH 0FFH FILLERS*/
13    2       DO I = X TO Y;                      /*DO LOOP TO PLACE THE BIT PATTERNS OF THE NUMBERS OR LETTERS*/
14    3       J = HEAD(Z+K);                      /*HEAD HAS THE CORRECT BIT PATTERNS FOR CONVERTING*/
15    3       DIS(I) = DIS(I) + SHL(J,8);         /*THE NUMBERS OR LETTERS ARE PLACED IN THE TOP HALF OF DISPLAY*/
16    3       K = K + 1;                          /*INCREMENT THE COUNTER FOR PROCEEDING THROUGH HEAD*/
17    3       END;
18    2       END PUT;

19    1       ASCII: PROCEDURE(X) PUBLIC;
20    2       DCL (FIRST,J,P,SECOND) BYTE, (X,Y) ADDRESS;

21    2       IF X < 351 THEN DO;                        /*PEAK OR CBH FOUND TO BE CONVERTED*/
23    3       NO = 0;                                    /*DUMMY VARIABLE FOR "NO" NOT PRINTED THIS SCAN*/
24    3       Y = HEIGHT(X);                             /*CONVERTING NUMBER OF 1/4 DEGREES TO NUMBER OF FEET*/
```

```
PL/M-80 COMPILER    RBC DISPLAY - ASCII

 25   3      FIRST,SECOND = 0;                   /*DUMMY VARIABLES TO SUPPRESS LED ZEROS*/
 26   3      P = (Y/1000);                       /*GET THOUSAND DIGIT*/
 27   3      IF P > 0 THEN DO;                   /*IF NON-ZERO PUT VALUE FROM HEAD (BIT PATTERN FOR THOUSAND
                                                 DIGIT) INTO ARRAY DIS*/
 29   4      CALL PUT(169,173,P);
 30   4      END;
 31   3      ELSE FIRST = 1;                     /*IF THOUSAND DIGIT ZERO;SUPPRESS AND SET DUMMY VARIABLE TO 1*/
 32   3      Y = Y - (P*1000);                   /*SUBTRACT THOSAND DIGIT FROM ORGINAL NUMBER */
 33   3      P = (Y/100);                        /*GET HUNDRED DIGIT*/
 34   3      IF P > 0 OR FIRST = 0 THEN DO;      /*IF HUNDRED DIGIT NONZERO OR THOUSAND DIGIT WAS NONZERO,
                                                 THEN GET BIT PATTERN OF NUMBER*/
 36   4      CALL PUT(175,179,P);
 37   4      END;
 38   3      ELSE SECOND = 1;                    /*IF THOUSAND AND HUNDRED DIGIT BOTH ZERO, SUPPRESS BOTH AND
                                                 SET DUMMY VARIABLE TO 1*/
 39   3      Y = Y - (P*100);                    /*SUBTRACT HUNDRED DIGIT OUT OF ORGINAL NUMBER*/
 40   3      P = (Y/10);                         /*GET TENS DIGIT*/
 41   3      IF P > 0 OR FIRST = 0 OR SECOND = 0 THEN DO;   /*IF ANY ONE OF THE THOUSANDS, HUNDREDS, OR TENS
                                                             DIGITS ARE NONZERO THEN PLACE TENS DIGIT IN ARRAY*/
 43   4      CALL PUT(181,185,P);
 44   4      END;
 45   3      P = Y - (P*10);                     /*SUBTRACT OUT TENS DIGIT FROM REMAINING NUMBER TO GET ONES DIGIT*/
 46   3      CALL PUT(187,191,P);                /* PLACE ONES DIGIT CONVERTED IN ARRAY REGARDLESS OF PAST OR
                                                 PRESENT DIGITS*/
 47   3      END;
 48   2      ELSE DO;                            /*NO PEAK OR NO CBH WAS FOUND SO PRINT LETTERS IN ARRAY*/
 49   3      NO = 1;                             /*DUMMY VARIABLE FOR "NO" PRINTED THIS SCAN*/
 50   3      DO J = 151 TO 168;                  /*DO LOOP TO CLEAR TOP HALF OF ARRAY TO PLACE NO CBH OR PEAK*/
 51   4      DIS(J) = DIS(J) AND 00FFH;          /*ZERO OUT TOP HALF OF ARRAY BUT LEAVE BOTTOM HALF*/
 52   4      END;
 53   3      CALL PUT(151,168,10);               /*PLACE "NO " IN ARRAY */
 54   3      IF X = 384 THEN CALL PUT(169,191,14);      /*PUT PEAK INTO DIS ARRAY ONE SCAN MODE*/
 56   3      ELSE CALL PUT(169,191,19);          /*PUT CBH INTO DIS ARRAY 5 SCAN MODE*/
 57   3      END;
 58   2      END ASCII;
 59   1      END ASCII$MOD;

MODULE INFORMATION:

     CODE AREA SIZE     = 0245H    581D
     VARIABLE AREA SIZE = 000FH     15D
     MAXIMUM STACK SIZE = 0006H      6D
     87 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - CHECK1

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE CHECK1MOD
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:CHECK1.SAV TITLE('RBC DISPLAY - CHECK1') NOOBJECT


             $PRINT(:LP:)  DEBUG

  1          CHECK1$MOD:  DO;

             /*THIS PROCEDURE TAKES THE 352 CBH DATA POINTS AND ELIMINATES ANY SHARP NOISE
             SPIKES, FILTERS AND AVERAGES THE POINTS (USING A 64 PART FILTER: 1 FOR I +- 3
             ,6 FOR I +-2, 15 FOR I +-1, AND 20 FOR CENTER VALUE), AND THEN FINDS THE
             MAXIMUM AND SECONDARY MAXIMUM PEAK FOR EACH SCAN (PEAK IS DEFINED AS FOUR
             INCREASING VALUES AND TWO DECREASING VALUES). THEN STORES HEIGHT IN DEGREES
             OF MAXIMUM AND SECONDARY MAXIMUM PEAK IN HEIT.  */

  2    1     DECLARE A(361) BYTE EXTERNAL, HEIT(20) ADDRESS EXTERNAL;
  3    1     DECLARE PEAK(80) BYTE EXTERNAL, PULSE(80) ADDRESS EXTERNAL, RCT(2) BYTE EXTERNAL;

  4    1     CHECK1: PROCEDURE(NO2) PUBLIC;

  5    2     DECLARE (DOWN,IX,J,NO2,Q,SVE1,UP) BYTE;
  6    2     DECLARE (I,P) ADDRESS;

  7    2     DO P = 21 TO 351;                              /*DO LOOP WHICH CHECKS FOR LARGE NOISE SPIKES*/
  8    3     I = (A(P-1) + A(P+1))/2;                       /*FIND AVERAGE OF TWO ADJACENT POINTS*/
  9    3     IF  DOUBLE(A(P)) >= 3 * I  THEN  A(P) = I;     /*IF CENTER POINT GREATER THAN 3 TIMES THE AVERAGE
                                                            /*THEN NOISE AND FILL WITH AVERAGE OF ADJACENT POINTS*/
 11    3     END;
 12    2     DO I = 24 TO 348;                              /*DO LOOP WHICH FILTERS AND AVERAGES*/
 13    3     P = A(I-3) + 6*(DOUBLE(A(I-2)) + A(I+2)) + 15*(DOUBLE(A(I-1))
              + A(I+1)) + 20*A(I) + A(I+3);  /*FILTER AND AVERAGE BY WEIGHTS 1,6,15,20,15,6,1FOR I-3,I-2,...,I+3*/
 14    3     A(I) = P/64;                     /*DIVIDE SUM BY THE TOTAL OF THE WEIGHTS*/
 15    3     END;
 16    2     DO I = 0 TO 79;                  /*ZERO OUT PEAK AND PULSE; PEAK WILL HAVE MAGNITUDE OF PEAK,
                                              AND PULSE WILL HAVE HEIGHT IN DEGREES OF PEAK*/
 17    3     PEAK(I),PULSE(I) = 0;
 18    3     END;
 19    2     DOWN,IX,UP = 0;                  /*DOWN = NUMBER OF DECREASING INTERVALS AFTER PEAK; IX = COUNTER
                                              FOR NUMBER OF PEAKS;UP = NUMBER OF INCREASING INTERVALS*/
 20    2     SVE1 = 0;                        /*SET MAXIMUM IN PEAK SO FAR TO ZERO*/
 21    2     DO I = 24 TO 347;                /*DO LOOP TO GO THROUGH CBH DATA POINTS TO FIND PEAKS*/
 22    3     IF IX < 79 THEN DO;              /*LIMIT NUMBER OF PEAKS TO 80*/
 24    4     IF A(I+1) > A(I) THEN DO;        /*CHECK FOR INCREASING INTERVALS*/
 26    5     IF DOWN = 1 THEN DO;             /*LAST POINT WAS A DECREASE, BUT THE POINT BEFORE THAT WAS NOT.
                                               NEEDED TWO DECREASES TO GET A PEAK, SO NEEDE TO CHECK IF
                                               SECOND PREVIOUS POINT WAS LESS THAN PRESENT*/
 28    6     IF SVE1 > A(I+1) THEN DO;        /*SVE1 IS THE SECOND PREVIOUS POINT. IF SVE1 IS LARGER THAN A(I+1)
                                              WE HAVE TWO DECREASING VALUES*/
 30    7     IF UP > 2 THEN DO;               /*CHECK NOW TO SEE IF 3 INCREASING INTERVALS OCUURRED BEFORE 2 DECREASING
                                              ;THIS DETERMINES A PEAK. IF PEAK FOUND DO FOLLOWING*/
 32    8     I = I + 1;                       /*INCREMENT I SO THAT POINT WILL NOT BE COUNTED AGAIN*/
 33    8     IX = IX + 1;                     /*INCREMENT PEAK COUNTER*/
 34    8     END;
 35    7     DOWN,UP = 0;                     /*EITHER A PEAK OR NOT, TWO DECREASING VALUES FOUND SO RESET DOWN AND UP*/
 36    7     END;
```

```
PL/M-80 COMPILER    RBC DISPLAY - CHECK1
  37   6        ELSE DO;                              /*SECOND PREVIOUS WAS NOT LARGER THAN CURRENT SO DISCARD
                                                       ONE DOWN, AND STILL SEEKING TOP OF PEAK*/
  38   7        DOWN = 0;                             /*SET DOWN BACK TO ZERO; DISCARD ONE DOWN VALUE*/
  39   7        UP = UP + 1;                          /*INCREMENT COUNTER FOR NUMBER OF INCREASING VALUES;STILL ON PEAK*/
  40   7        SVE1 = A(I+1);                        /*STORE AWAY CURRENT VALUE; MAXIMUM IN PEAK SO FAR*/
  41   7        IF UP > 2 THEN DO;                    /*THREE INCREASING VALUES NECESSARY FOR A PEAK*/
  43   8        PEAK(IX) = A(I+1);                    /*PEAK HAS THE VALUE OF THE LARGEST DIGITAL RETURN FOR THIS PEAK*/
  44   8        PULSE(IX) = I + 1;                    /*PULSE HAS THE HEIGHT IN 1/4DEGREES OF THE PEAK*/
  45   8        END;  END;  END;
  48   5        ELSE DO;                              /*THE PRECEEDING POINT WAS NOT A DECREASING ONE*/
  49   6        UP = UP + 1;                          /*INCREMENT COUNTER FOR NUMBER OF INCREASING VALUES*/
  50   6        SVE1 = A(I+1);                        /*STORE AWAY CURRENT VALUE; MAXIMUM IN PEAK SO FAR*/
  51   6        IF UP > 2 THEN DO;                    /*3 INCREASING INTERVALS NECESSARY FOR A PEAK*/
  53   7        PEAK(IX) = A(I+1);                    /*PEAK HAS THE MAGNITUDE OF THE LARGEST DIGITAL RETURN*/
  54   7        PULSE(IX) = I + 1;                    /*PULSE HAS THE HEIGHT IN 1/4 DEGREES OF THE PEAK*/
  55   7        END;  END;  END;
  58   4        ELSE DO;                              /*CURRENT VALUE IS LESS THAN OR EQUAL TO THE PREVIOUS VALUE*/
  59   5        IF DOWN = 0 THEN DOWN = 1;            /*FIRST DECREASING INTERVAL AFTER AN INCREASING ONE*/
  61   5        ELSE DO;                              /*SECOND DECREASING INTERVAL AFTER AN INCREASING ONE*/
  62   6        IF UP > 2 THEN DO;                    /*A PEAK IS FOUND; 3 INCREASING INTERVALS AND 2 DECREASING ONES*/
  64   7        I = I + 1;                            /*INCREMENT DO LOOP COUNTER SO POINT WILL NOT BE USED TWICE*/
  65   7        IX = IX + 1;                          /*INCREMENT COUNTER FOR PEAK COUNT*/
  66   7        END;
  67   6        DOWN,UP = 0;                          /*SECOND DECREASING INTERVAL, SO RESET UP AND DOWN, PEAK OR NOT*/
  68   6        END;  END;  END;  END;
  72   2        IF IX > 0 THEN DO;                    /*IF IX IS GREATER THAN 0,PEAK(S) HAVE BEEN FOUND*/
  74   3        Q = 1;                                /*DUMMY VARIABLE FOR REMAINING OR LEAVING THE FOLLOWING DO LOOP*/
  75   3        DO WHILE Q;                           /*DO WHILE LOOP TO ORDER ANY PEAKS FOUND, LARGESTPEAK AT PEAK(0)*/
  76   4        Q = 0;                                /*SET DUMMY VARIABLE TO ZERO WHICH WILL CAUSE AN EXIT FROM LOOP
                                                      UNLESS A PEAK IS FOUND TO BE OUT OF ORDER */
  77   4        DO I = 0 TO IX-1;                     /*DO LOOP TO GO THROUGH NUMBER OF PEAKS - 1*/
  78   5        IF PEAK(I) < PEAK(I+1) THEN DO;         /*A LARGER PEAK FOUND IN A HIGHER POSITION*/
  80   6        Q = 1;                                /*DUMMY VARIABLE TO SAY FOUND ANOTHER PEAK OUT OFORDER*/
  81   6        J = PEAK(I);                          /*CHANGE THE TWO PEAKS THAT ARE OUT OF ORDER INTO THE CORRECT ORDER*/
  82   6        P = PULSE(I);
  83   6        PEAK(I) = PEAK(I+1);
  84   6        PULSE(I) = PULSE(I+1);
  85   6        PEAK(I+1) = J;
  86   6        PULSE(I+1) = P;
  87   6        END;  END;  END;  END;
  91   2        J = (RCT(NO2) - 1) + (10 * NO2);  /*FINDING THE CORRECT PLACE IN HEIT TO STORE THE CURRENT SCAN*/
  92   2        HEIT(J),HEIT(J+5) = 1000;         /*PRESET THE CORRECT STORAGE PLACE FOR MAX AND 2ND MAX
                                                  TO 1000 WHICH REPRESENTS NO CBH*/
  93   2        DO I = 0 TO 1;                    /*DO LOOP TO STORE THE TWO MAXIMUMS*/
  94   3        IF IX > I THEN DO;                /*CHECK TO SEE IF THERE ARE ANY PEAKS; IF SO HOW MANY(1 OR 2)*/
  96   4        HEIT(J+5*I) = PULSE(I);           /*STORE HEIGHT IN 1/4 DEGREES OF 1ST AND 2ND MAXS*/
  97   4        END;  END;
  99   2        END CHECK1;
 100   1        END CHECK1$MOD;


MODULE INFORMATION:

     CODE AREA SIZE     = 03B5H    949D
     VARIABLE AREA SIZE = 000BH     11D
     MAXIMUM STACK SIZE = 0004H      4D
     100 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - DATA CHECK

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE DCMOD
OBJECT MODULE PLACED IN :F1:DC.OBJ
COMPILER INVOKED BY:  PLM80 :F1:DC.SRC TITLE('RBC DISPLAY - DATA CHECK')


              $PRINT(:LP:)  DEBUG
   1          DC$MOD:  DO;

   2   1      MTIME: PROCEDURE(X) EXTERNAL;
   3   2      DECLARE X BYTE;
   4   2      END MTIME;

   5   1      DECLARE J BYTE EXTERNAL;

   6   1      DATA$CHECK: PROCEDURE (P,Y,Z) PUBLIC REENTRANT;

              /*PROCEDURE IS A FLAG CHECKER. THE PROGRAM REMAINS IN THIS PROCEDURE FOR 100MS
              OR UNTIL A TRUE CONDITION EXISTS. J IS AN IDENTIFIER WHICH CAN BE CHECKED TO
              DETERMINE THE REASON FOR EXIT ( J=0 TRUE CONDITION;J = 10 100MS EXIT)*/

   7   2      DECLARE (Y,Z) BYTE, (I,P) ADDRESS;
   8   2      J = 0;                                      /*SET IDENTIFER TO ZERO*/
   9   2      DO I = 0 TO 847;                            /*DO LOOP TO CHECK CONDITION AND TO WASTE TIME IN BETWEEN*/
  10   3      IF (MEMORY(P) AND Y) = Z THEN RETURN;       /*CHECK FOR CONDITION TRUE, IF TRUE RETURN*/
  12   3      CALL MTIME(1);                              /*TOTAL WAIT OF 847 X 118MICRO SEC = APPROX 100M SEC*/
  13   3      END;
  14   2      J = 10;                                     /*FINISHED DO LOOP WITHOUT A TRUE CONDITION, SET J*/
  15   2      END DATA$CHECK;
  16   1      END DC$MOD;


MODULE INFORMATION:

     CODE AREA SIZE     = 0069H    105D
     VARIABLE AREA SIZE = 0000H      0D
     MAXIMUM STACK SIZE = 000AH     10D
     24 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - DETOFF

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE DETOFFMOD
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:DETOFF.SAV TITLE('RBC DISPLAY - DETOFF') NOOBJECT


             $PRINT(:LP:)  DEBUG

  1          DETOFF$MOD:  DO;

  2   1      DECLARE (K6,K7,OFFS) BYTE EXTERNAL;
  3   1      DECLARE C(20) BYTE EXTERNAL, OCT(2) BYTE EXTERNAL, OFFSET(2) BYTE EXTERNAL;

  4   1      DETOFF: PROCEDURE PUBLIC;

             /*PROCEDURE TAKES THE ONE SCAN AVERAGE OF DATA COLLECTED DURING THE OFF SCAN
             AND FINDS THE AVERAGE FOR THE LAST SCANS(UP TO A MAXIMUM OF 10), STORES IN OFFSET(0)
             FOR LAMP 1 OR OFFSET(1) FOR LAMP 2*/

  5   2      DECLARE (I,P) ADDRESS;

  6   2      P = OFFS;                     /*OFFS IS AVERAGE NOISE FROM ONE SCAN*/
  7   2      IF P > 3CH THEN P = 3CH;      /*ONLY 4 SIG BITS FOR OFFSET, BITS 2,3,4,5*/
  9   2      C(OCT(K6)+K7) = P;            /*OFFSET IS CALCULATED AFTER ZERO DEGREE SWITCH FOR NEXT LIGHT, THEREFORE
                                           K6 IS OPPOSITE OF K3(0 FOR LAMP 1 AND 1 FOR LAMP 2);
                                           K7 IS USED TO STORE THE TEN CURRENT OFFS IN C, THEREFORE K7 IS EITHER
                                           0 OR 10 DEPENDING ON LAMP*/
 10   2      P = 0;                        /*P SET TO ZERO TO BE USED FOR TOTAL OF PAST OFFS UPTO 10*/
 11   2      DO I = 0 TO OCT(K6);          /*DO LOOP TO SUM ALL OFFS*/
 12   3      P = P + C(I+K7);
 13   3      END;
 14   2      IF OCT(K6) > 0 THEN P = P/OCT(K6);           /*AVERAGE OF PAST OFFS*/
 16   2      OFFSET(K6) = SHR(P,2) + 2;                   /*ONLY 4 SIG BITS, BUT MUST BE IN BITS 0,1,2,3;
                                                         ADD 2 OFFSET TO CALCULATED VALUE TO REDUCE NOISE*/
 17   2      IF OFFSET(K6) > 0FH THEN OFFSET(K6) = 0FH;  /*LARGEST OFFSET CAN BE IS 4 BITS ON*/
 19   2      IF OCT(K6) > 8 THEN DO;                      /*IF ALREADY HAVE TEN OFFS MOVE THEN TENTH OUT AND
                                                         MAKE ROOM FOR THE NEXT ONE */
 21   3      OCT(K6) = 8;
 22   3      DO I = 0 TO 9;
 23   4      C(I+K7) = C(I+K7+1);                         /*MOVE 1 TO 0, 2 TO 1 ETC*/
 24   4      END;  END;
 26   2      OCT(K6) = OCT(K6) + 1;                       /*ADD ONE TO THE COUNT OF OFFS*/
 27   2      END DETOFF;
 28   1      END DETOFF$MOD;


MODULE INFORMATION:

     CODE AREA SIZE      = 0120H    288D
     VARIABLE AREA SIZE  = 0004H      4D
     MAXIMUM STACK SIZE  = 0004H      4D
     38 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - FILL

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE FILLMOD
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:FILL.SAV TITLE('RBC DISPLAY - FILL') NOOBJECT


                $PRINT(:LP:)  DEBUG

  1             FILL$MOD:  DO;

  2    1        ASCII: PROCEDURE(X) EXTERNAL;
  3    2        DECLARE X ADDRESS;
  4    2        END ASCII;

  5    1        DECLARE (CURSOR,K3,SCAN) BYTE EXTERNAL;
  6    1        DECLARE BEIL(2) ADDRESS EXTERNAL, DIS(200) ADDRESS EXTERNAL;

  7    1        FILL: PROCEDURE PUBLIC;

                /*THIS PROCEDURE CAUSES THE CURSOR TO BE REVERSE VIDEO, CLEANS OUT THE SPACE AT
                THE END OF THE DISPLAY FOR THE NUMBERS TO BE DISPLAYED, CALLS ASCII WHICH CONVERTS
                THE PEAK OR CBH TO A NUMBER AND PUTS INTO DISPLAY ARRAY, WRITES THE ARRAY INTO
                THE MEMORY FOR THE DISPLAY UNIT, AND THEN GIVES THE SWITCH SIGNAL TO UPDATE
                THE DISPLAY*/

  8    2        DECLARE I ADDRESS;

  9    2        DO I = 176 TO 192;                  /*DO LOOP TO CLEAN OUT MEMORY AT THE END WHERE NO DATA IS STORED*/
 10    3        DIS(I) = 00H;
 11    3        END;
 12    2        DIS(CURSOR) = NOT DIS(CURSOR);      /*REVERSES THE BIT IN MEMORY WHERE THE CURSOR WILL BE STORED
                                                    CAUSING THE CURSOR TO BE REVERSE VIDEO*/
 13    2        DO I = 169 TO 175;                  /* CLEANS OUT THE TOP PART OF DISPLAY WHERE THE NUMBER WILL
                                                    BE STORED. ONLY THE TOP HALF BECAUSE DATA MIGHT ALSO BE HERE*/
 14    3        DIS(I) = DIS(I) AND 00FFH;
 15    3        END;
 16    2        SCAN = MEMORY(6601H) AND 03H;       /*TEST TO SEE IF IN ONE SCAN OR 5 SCAN MODE, OR MANUAL*/
 17    2        IF SCAN > 0 THEN CALL ASCII(CURSOR*2);   /*IF IN ONE SCAN OR MANUAL MODE PRINT NUMBER FOR CURSOR*/
 19    2        ELSE CALL ASCII(BEIL(K3));          /*IF IN FIVE SCAN MODE PRINT CBH HEIGHT*/
 20    2        DO I = 0 TO 192;                    /*DO LOOP TO PUT DISPLAY UPDATE INTO DISPLAY MEMORY*/
 21    3        MEMORY(6001H+I) = LOW(DIS(I));      /*BOTTOM HALF OF DISPLAY*/
 22    3        MEMORY(6202H+I) = HIGH(DIS(I));     /*TOP HALF OF DISPLAY*/
 23    3        END;
 24    2        CALL TIME(20);
 25    2        MEMORY(6400H) = 00H;                /*UPDATE SWITCH TO CAUSE DISPLAY MEMORIES TO BE REVERSED
                                                    THEREFORE DISPLAY CURRENT SCAN*/

 26    2        END FILL;
 27    1        END FILL$MOD;


MODULE INFORMATION:

     CODE AREA SIZE     = 00F1H    241D
     VARIABLE AREA SIZE = 0002H      2D
     MAXIMUM STACK SIZE = 0002H      2D
     42 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - GAIN

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE GAINMOD
OBJECT MODULE PLACED IN :F1:GAIN.OBJ
COMPILER INVOKED BY:  PLM80 :F1:GAIN.SRC TITLE('RBC DISPLAY - GAIN')


                 $ PRINT(:LP:)     DEBUG

   1             GAIN$MOD:  DO;

   2    1        DECLARE GAIN(2) BYTE EXTERNAL;

   3    1        GAINER: PROCEDURE(X,Y) PUBLIC;

                 /*THIS PROCEDURE CALCULATES THE GAIN IF ANY NEEDED TO BRING THE START BANG UP
                 TO FULL SCALE. THE MAXIMUM VALUE IN THE FIRST 5 DEGREES OF THE SCAN IS THE VARIABLE
                 X AND Y IS THE LAMP THIS VALUE APPLIES TO. THE FULL SCALE VALUE (255) IS DIVIDED BY
                 THE MAXIMUM SIGNAL OBTAINED. ONLY FOUR DISCREET GAINS ARE POSSIBLE:

                           0        1.00
                           1        1.33
                           2        1.67
                           3        2.00

                 THIS GAIN IS THEN APPLIED TO THE CORRESPONDING LAMP FOR THE NEXT FIVE SCANS.*/

   4    2        DECLARE (X,Y) BYTE, P ADDRESS;
   5    2        P = 25500 / X;                 /*DIVIDE FULL SCALE(255) * 100 BY LARGEST VALUE FROM 0-5DEGREES*/
   6    2        X = 0;                         /*GAIN OF 1.00 IS ASSUMED UNLESS CHANGED BELOW*/
                                                /*CALCULATE HOW MUCH IF ANY GAIN IS NEEDED*/
   7    2        IF P > 133 THEN X = 1;         /*GAIN OF 1.33 IS NEEDED*/
   9    2        IF P > 167 THEN X = 2;         /*GAIN OF 1.67 IS NEEDED*/
  11    2        IF P > 200 THEN X = 3;         /*GAIN OF 2.00 IS NEEDED*/
  13    2        GAIN(Y) = X;                   /*PLACE GAIN NEEDED INTO GAIN FOR APPROPRIATE LAMP*/
  14    2        END GAINER;
  15    1        END GAIN$MOD;


MODULE INFORMATION:

     CODE AREA SIZE     = 0058H     88D
     VARIABLE AREA SIZE = 0004H      4D
     MAXIMUM STACK SIZE = 0004H      4D
     30 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - HEIGHT

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE HEIGHTMOD
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:HEIGHT.SAV TITLE('RBC DISPLAY - HEIGHT') NOOBJECT


                $PRINT(:LP:) DEBUG
   1            HEIGHT$MOD:  DO;
   2   1        DECLARE DCL LITERALLY 'DECLARE';

   3   1        HEIGHT: PROCEDURE(Y) ADDRESS PUBLIC;

                /*THIS PROCEDURE CONVERTS THE HEIGHT OF THE PEAKS IN 1/4 DEGREES TO HEIGHT IN
                FEET*/

   4   2        DCL Y ADDRESS;

                /*VALUES GO FROM 0.00 TO 88.25 DEGREES AT 1/4 DEGREE INTERVALS BUT FOR CBH READINGS EXPECT VALUES FROM
                6.75 TO 86.5 DEGREES ONLY BECAUSE OF FILTERING,AVERAGING, AND DEFINITION OF PEAK*/

   5   2        DCL HIT(354) ADDRESS DATA(0,2,3,5,7,9,10,12,14,16,17,
                19,21,23,24,26,28,30,31,33,35,37,39,40,42,44,46,47,49,51,53,
                54,56,58,60,62,63,65,67,69,71,72,74,76,78,80,81,83,85,87,89,
                91,92,94,96,98,100,102,103,105,107,109,111,113,115,117,118,120,122,124,126,
                128,130,132,134,136,138,140,142,144,146,148,150,152,154,156,158,160,162,164,166,
                168,170,172,174,176,178,180,182,184,187,189,191,193,195,197,199,202,204,206,208,
                210,213,215,217,219,222,224,226,229,231,233,235,238,240,243,245,248,250,252,255,
                257,260,262,265,267,270,272,275,278,280,283,285,288,291,293,296,299,302,304,307,
                310,313,315,318,321,324,327,330,333,336,339,342,345,348,351,354,357,360,363,367,
                370,373,376,380,383,386,390,393,397,400,404,407,411,414,418,422,425,429,433,437,
                440,444,448,452,456,460,464,468,472,478,481,485,490,494,498,503,507,512,517,521,
                526,531,536,541,546,551,556,561,566,571,577,582,587,593,599,604,610,616,622,628,
                634,640,646,653,659,666,672,679,686,693,700,707,714,722,729,737,744,752,760,768,
                777,785,794,802,811,820,829,839,848,858,868,878,888,898,909,920,931,942,954,966,
                978,990,1003,1015,1029,1042,1056,1070,1084,1099,1114,1130,1145,1162,1178,1195,1213,1231,1250,1269,
                1288,1308,1329,1350,1372,1395,1418,1442,1467,1493,1519,1547,1575,1604,1635,1666,1699,1733,1768,1804,
                1842,1882,1923,1966,2011,2058,2107,2158,2212,2268,2328,2390,2456,2525,2599,2676,2759,2846,2939,3038,
                3144,3259,3380,3510,3652,3806,3972,4155,4353,4572,4814,5082,5383,5720,6103,6540,7044,7632,8328,9162,
                10181,11455,13092);

   6   2        RETURN HIT(Y);
   7   2        END HEIGHT;
   8   1        END HEIGHT$MOD;


MODULE INFORMATION:

     CODE AREA SIZE     = 02D7H    727D
     VARIABLE AREA SIZE = 0002H      2D
     MAXIMUM STACK SIZE = 0000H      0D
     37 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - INTONE

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE INTONEMOD
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:INTONE.SAV TITLE('RBC DISPLAY - INTONE') NOOBJECT


              $PRINT(:LP:) DEBUG

  1           INTONE$MOD:  DO;

  2   1       DECLARE (GETOFF,K3,K6,K7,RFLAG) BYTE EXTERNAL, IC ADDRESS EXTERNAL;

  3   1       INTONE: PROCEDURE PUBLIC;

              /*THIS PROCEDURE HANDLES THE INTERRUPT ASSOCIATED WITH THE ZERO DEGREE SWITCH FOR EACH LAMP.
              IT FIRST SET THE COUNTER TO START THE 120HZ INTERRUPTS, THEN CHANGES THE DUMMY
              VARIABLES TO SIGNIFY WHAT LAMP IS BEING PROCESSED*/

  4   2       IF IC <> 362 THEN RETURN;    /*THIS IS DONE TO PREVENT DOUBLE OR TRIPLE INTERRUPTS CAUSED
                                           BY A LONG ZERO DEGREE SWITCH CLOSURE FROM INTERRUPTING MORE THAN ONCE*/
  6   2       MEMORY(8403H) = 30H;         /*LOADING THE 8253 COUNTER #0 TO START COLLECTING OF DATA*/
  7   2       MEMORY(8400H) = 01H;         /*LS BYTE LOADED INTO 8253; A ONE IS USED TO GET AN INTERRUPT WITHIN
                                           50 MICRO SECS OF START. THIS AS CLOSE TO ZERO AS NECESSARY*/
  8   2       MEMORY(8400H) = 00H;         /*MS BYTE LOADED INTO 8253*/
  9   2       IF RFLAG = 4 THEN DO;        /*RFLAG = 4 REPRESENTS CBH DATA FROM LAMP2 HAS BEEN PROCESSED AND
                                           NOW WAITING FOR ZERO DEGREE SWITCH ON LAMP1*/
 11   3       K3 = 0;                      /*DUMMY VARIABLE K3 = 0 LAMP 1*/
 12   3       RFLAG,K6 = 1;                /*RFLAG = 1 REPRESENTS CBH DATA BEING COLLECTED FROM LAMP 1;
                                           K6 IS A DUMMY VARIABLE WHICH IS THE COMPLEMENT OF K3. THIS IS USED
                                           WHEN THE OFFSET IS CALCULATED TO PUT THE OFFSET INTHE RIGHT
                                           POSITION SO IT IS USED FOR THE CORRECT LAMP*/
 13   3       K7 = 10;                     /*K7 IS A DUMMY VARIABLE USED IN OFFCAL SO THE AVERAGE NOISE FROM
                                           EACH SCAN IS STORED IN THE CORRECT LOCATION FOR THE
                                           LAMP IT REPRESENTS*/
 14   3       END;
 15   2       ELSE DO;                     /*RFLAG = 2 REPRESENTS CBH DATA FROM LAMP1 HAS BEEN PROCESSED AND
                                           NOW WAITING FOR ZERO DEGREE SWITCH ON LAMP 2*/
 16   3       K3 = 1;                      /*DUMMY VARIABLE K3 = 1 LAMP 2*/
 17   3       RFLAG = 3;                   /*RFLAG = 3 REPRESENTS COLLECTING CBH DATA FROM LAMP 2*/
 18   3       K6,K7 = 0;                   /*SEE ABOVE FOR DESCRIPTON OF K6 AND K7*/
 19   3       END;
 20   2       MEMORY(6602H) = 0;          /*SET GAIN CORRECTION TO ZERO FOR PRESENT LAMP TO COLLECT NEW GAIN DATA*/
 21   2       GETOFF,IC = 0;               /*GETOFF=0 SIGNALS END OF COLLECTING DATA FOR OFFSET; IC IS THE
                                           COUNTER FOR INCOMING CBH DATA POINTS*/
 22   2       END INTONE;
 23   1       END INTONE$MOD;


MODULE INFORMATION:

     CODE AREA SIZE     = 005EH     94D
     VARIABLE AREA SIZE = 0000H      0D
     MAXIMUM STACK SIZE = 0002H      2D
     40 LINES READ
     0 PROGRAM ERROR(S)
END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - INTTWO

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE INTTWOMOD
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:INTTWO.SAV TITLE('RBC DISPLAY - INTTWO') NOOBJECT


                 $PRINT(:LP:)  DEBUG

   1             INTTWO$MOD:  DO;

   2   1         DATA$CHECK: PROCEDURE(P,Y,Z) EXTERNAL;
   3   2         DECLARE (Y,Z) BYTE, P ADDRESS;
   4   2         END DATA$CHECK;

   5   1         OFFCAL: PROCEDURE EXTERNAL;
   6   2         END OFFCAL;

   7   1         DECLARE (CNT,COFF,DONE,K3,MXCT,MXOFF1,MXOFF2,MXPK,RFLAG) BYTE EXTERNAL;
   8   1         DECLARE (IC,POFF) ADDRESS EXTERNAL;
   9   1         DECLARE A(361) BYTE EXTERNAL, GAIN(2) BYTE EXTERNAL, OFFSET(2) BYTE EXTERNAL;
  10   1         DECLARE PEAKO(200) BYTE EXTERNAL;

  11   1         INTTWO: PROCEDURE PUBLIC;

                 /*THIS PROCEDURE PROCESSES THE 120HZ INTERRUPT EITHER FOR COLLECTING CBH DATA
                 OR FOR COLLECTING NOISE DATA FOR THE OFFSET CORRECTION. IT FIRST RESET THE
                 COUNTER USED TO PROVIDE THE 120HZ INTERRUPT, AND THEN CHECKS TO SEE IF CBH OR
                 NOISE DATA ARE BEING COLLECTED. IF NOISE DATA ARE BEING COLLECTED OFFCAL IS
                 CALLED. OTHERWISE THE PROGRAM HERE IS EXECUTED. FOR CBH DATA, THE DATA IS READ
                 FROM THE A TO D CONVERTER AND STORED IN ARRAY A. THEN IF THE SCAN IS AT 5
                 DEGREES THE GAIN AND OFFSET CORRECTIONS ARE APPLIED. IF THE SCAN IS AT 88
                 DEGREES THE OFFSET CORRECTION IS TAKEN OUT AND CONDITIONS ARE SET TO PROCESS
                 THE CBH DATA JUST COLLECTED AND TO BEGIN COLLECTED OFFSET DATA.  */

  12   2         MEMORY(8403H) = 30H;            /*LOAD 8253 COUNTER #1 FOR USE AS THE 120HZ INTERRUPT*/
  13   2         MEMORY(8400H) = 9DH;            /*LOAD LS BYTE FIRST; USING 157 BECAUSE INPUT CLOCK TO COUNTER
                                                 IS 50 MICRO SECS AND ABOUT 150 MICRO SECS ARE USED PROCESSING
                                                 THE INTERRUPT ITSELF*/
  14   2         MEMORY(8400H) = 00H;            /*LOAD MS BYTE SECNOD*/
  15   2         IF RFLAG = 2 OR RFLAG = 4 THEN CALL OFFCAL;    /*CBH DATA ALREADY COLLECTED SO USE OFF SCAN
                                                 TO COLLECT NOISE DATA TO BE USED FOR OFFSET CORRECTION*/
  17   2         ELSE DO;                        /*COLLECT CBH DATA; RFLAG = 1 OR 3*/
  18   3         A(IC) = MEMORY(6800H);          /*A READ OF THIS POSITION BEGINS THE CONVERT OF THE DATA*/
  19   3         CALL DATA$CHECK(6602H,20H,00H); /*THIS WAITS FOR END OF CONVERSION FLAG*/
  20   3         A(IC) = MEMORY(6800H);          /*READ AND STORE AWAY CONVERTED DATA 0 - 255*/
  21   3         IC = IC + 1;                    /*INCREASE COUNTER FOR CBH DATA POINTS*/
  22   3         IF IC = 20 THEN DO;             /*LOOK FOR START BAND ONLY IN THE FIRST 4.75 DEGREES; THIS DATA WILL
                                                 BE USED FOR THE GAIN CORRECTION*/
  24   4         MEMORY(6602H) = GAIN(K3);       /*FOR THE DEGREES AFTER 4.75 GAIN AND OFFSET CORRECTION ARE APPLIED*/
  25   4         MEMORY(6600H) = OFFSET(K3);
  26   4         END;
  27   3         IF IC > 360 THEN DO;            /*ONLY COLLECT CBH DATA UP TO 90 DEGREES; AFTER THAT PROCESS
                                                 THE DATA AND START COLLECTING NOISE DATA FROM OFF SCAN FOR OFFSET*/
  29   4         RFLAG = RFLAG + 1;              /*CHANGE RFLAG FROM 1 TO 2 OR FROM 3 TO 4*/
  30   4         MEMORY(6600H) = 00H;            /*TAKE OUT OFFSET CORRECTION; BECAUSE NOISE VALUES WILL BE READ
                                                 TO DETERMINE NEW OFFSET*/
  31   4         DONE = 0;                       /*DUMMY VARIABLE DONE =0 CBH DATA TO BE PROCESSED*/
```

```
 32   4      COFF,MXOFF1,MXOFF2 = 0;          /*COFF,MXOFF1, AND MXOFF2 ARE USED IN OFFCAL; COFF IS THE COUNTER
                                              FOR THE NOISE PEAKS FOUND IN THE OFF SCAN, AND MXOFF1 AND MXOFF2
                                              ARE THE MAXIMUM AND 2ND MAXIMUM PEAKS FOUND IN THE OFFSCAN.
                                              MXOFF1 AND 2 ARE USED TO DETERMINE NO CBH ON A SUNNY DAY WHEN
                                              THE SIGNAL CBH DATA IS VERY NOISY*/
 33   4      POFF = 1;                        /*POFF IS USED IN OFFCAL TO SPECIFY WHETHER THE LAST VALUE
                                              WAS A PEAK OR NOT;POFF =1, LAST WAS NOT A PEAK*/
 34   4      DO CNT = 0 TO 199;               /*DO LOOP TO ZERO OUT PEAKO WHICH IS USED IN OFFCAL TO STORE
                                              DATA COLLECTED  IN THE OFF SCAN*/
 35   5      PEAKO(CNT) = 0;
 36   5      END;  END;  END;
 39   2      END INTTWO;
 40   1      END INTTWO$MOD;

MODULE INFORMATION:

     CODE AREA SIZE     = 00C5H    197D
     VARIABLE AREA SIZE = 0000H      0D
     MAXIMUM STACK SIZE = 0004H      4D
     64 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - MTIME

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE MTIMEMOD
OBJECT MODULE PLACED IN :F1:MTIME.OBJ
COMPILER INVOKED BY:  PLM80 :F1:MTIME.SRC TITLE('RBC DISPLAY - MTIME')


              $PRINT(:LP:)  DEBUG
   1          MTIME$MOD:  DO;

   2    1     MTIME: PROCEDURE(X) PUBLIC REENTRANT;

              /*THE PURPOSE OF THIS PROCEDURE IS TO SIMPLY WASTE TIME. CALLING MTIME WITH A
              ZERO IS ILLEGAL. CALLING MTIME WITH A ONE RESULTS IN A WAIT OF 118 MICRO SECS*/

   3    2     DECLARE (X,J) BYTE;

   4    2     LOOP:  J = 2;                        /*DOES THIS LOOP X TIMES WHEN CALLED*/
   5    2     LOOP1: J = J - 1;                    /*DOES THIS LOOP 2 TIMES FOR EVERY X TIMES*/
   6    2     IF J <> 0 THEN GO TO LOOP1;
   8    2     X = X - 1;
   9    2     IF X <> 0 THEN GO TO LOOP;

  11    2     END MTIME;
  12    1     END MTIME$MOD;


MODULE INFORMATION:

     CODE AREA SIZE     = 0028H     40D
     VARIABLE AREA SIZE = 0000H      0D
     MAXIMUM STACK SIZE = 0002H      2D
     18 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - OFFCAL

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE OFFCALMOD
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:OFFCAL.SAV TITLE('RBC DISPLAY - OFFCAL') NOOBJECT


              $PRINT(:LP:)  DEBUG
  1           OFFCAL$MOD: DO;

  2    1      DATA$CHECK: PROCEDURE(P,Y,Z) EXTERNAL;
  3    2      DECLARE (Y,Z) BYTE, P ADDRESS;
  4    2      END DATA$CHECK;

  5    1      DECLARE PEAKO(200) BYTE EXTERNAL;
  6    1      DECLARE (COFF,MXOFF1,MXOFF2) BYTE EXTERNAL, POFF ADDRESS EXTERNAL;

  7    1      OFFCAL: PROCEDURE PUBLIC;

              /*THIS PROCEDURE IS USED BY THE 120HZ INTERRUPT DURING THE OFF SCAN TO COLLECT
              NOISE DATA TO BE USED TO CALCULATE THE OFFSET NECESSARY. IT CONVERTS THE DATA TO
              A DIGITAL VALUE FIRST THEN FINDS THE NOISE PEAKS (UP TO A MAXIMUM OF 200) IN THIS SCAN.
              A NOISE PEAK IS DEFINED AS A NON-ZERO VALUE FOLLOWED BY A LESSER VALUE.
              ALSO THE LARGEST AND SECOND LARGEST NOISE PEAKS ARE FOUND. THROUGH PAST EXPERIENCE
              ON SUNNY DAYS A LARGE AMOUNT OF NOISE IS FOUND. THESE TWO MAXIMUMS ARE USED LATER
              TO DIFFERENTIATE BETWEEN LARGE NOISE RETURNS ON THE CBH DATA SCAN AND REAL CLOUD
              RETURNS.
              IF THESE TWO MAXIMUMS ARE OVER A QUARTER SCALE THEN A SUNNY DAY IS ASSUMED AND
              NO CBH IS REPORTED FOR THE ONE MINUTE READING. */

  8    2      DECLARE X BYTE;

  9    2      X = MEMORY(6800H);                /*A READ OF THIS POSITION BEGINS CONVERT*/
 10    2      CALL DATA$CHECK(6602H,20H,00H);   /*WAIT FOR END OF CONVERT*/
 11    2      X = MEMORY(6800H);                /*READ THE CONVERT DIGITAL VALUE*/
 12    2      IF X > PEAKO(COFF) THEN DO;       /*CURRENT GREATER THAN PEAK VALUE*/
 14    3      PEAKO(COFF) = X;                  /*STORE CURRENT VALUE IN PEAK POSITION*/
 15    3      POFF = 0;                         /*IDENTIFIER FOR LAST VALUE WAS A PEAK*/
 16    3      END;
 17    2      ELSE DO;                          /*CURRENT VALUE NOT GREATER THAN PEAK VALUE*/
 18    3      IF POFF = 0 THEN DO;              /*PRECEEDING VALUE WAS A PEAK THEN DO*/
 20    4      POFF = 1;                         /*PRECEEDING VALUE NOT A PEAK*/
 21    4      IF PEAKO(COFF) > MXOFF1 THEN MXOFF1 = PEAKO(COFF);  /*LARGEST PEAK OF OFF SCAN*/
 23    4      ELSE DO;                          /*NOT THE LARGEST PEAK OF OFF SCAN, BUT CHECK FOR SECOND LARGEST*/
 24    5      IF PEAKO(COFF) > MXOFF2 THEN MXOFF2 = PEAKO(COFF);  /*SECOND LARGEST PEAK OF OFFSCAN*/
 26    5      END;
 27    4      IF COFF < 199 THEN COFF = COFF + 1;              /*INCREMENT PEAK COUNT BY ONE IF LESS THAN 199*/
 29    4      END;
 30    3      ELSE POFF = 1;                    /*LAST VALUE NOT A PEAK*/
 31    3      END;
 32    2      END OFFCAL;
 33    1      END OFFCAL$MOD;

MODULE INFORMATION:

     CODE AREA SIZE     = 00A3H    163D
     VARIABLE AREA SIZE = 0001H      1D
     MAXIMUM STACK SIZE = 0004H      4D
     45 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - RESET

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE RESETMOD
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:RESET.SAV TITLE('RBC DISPLAY - RESET')  NOOBJECT


                 $PRINT(:LP:)  DEBUG

  1              RESET$MOD:  DO;

  2    1         DECLARE DONE BYTE EXTERNAL, IC ADDRESS EXTERNAL;

  3    1         RESET: PROCEDURE PUBLIC;

                 /*THIS PROCEDURE RESETS THE NECESSARY VARIABLES AFTER THE CURRENT SCAN HAS BEEN
                 PROCESSED*/

  4    2         IF DONE = 0 THEN DO;     /*DONE = 0 MEANS A SCAN HAS FINISHED BUT HAS NOT BEEN FULLY PROCESSED*/
  6    3         DONE = 1;                /*DONE = 1 MEANS LAST SCAN HAS BEEN PROCESSED AND WAITING FOR NEW SCAN*/
  7    3         IC = 362;                /*PROCESSING OF PREVIOUS LAMP IS DONE; THEREFORE REENABLE INTER ONE*/
  8    3         END;

  9    2         END RESET;
 10    1         END RESET$MOD;


MODULE INFORMATION:

     CODE AREA SIZE     = 0014H     20D
     VARIABLE AREA SIZE = 0000H      0D
     MAXIMUM STACK SIZE = 0000H      0D
     18 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - SCAN5

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE SCAN5MOD
NO OBJECT MODULE REQUESTED
COMPILER INVOKED BY:  PLM80 :F1:SCAN5.SAV TITLE('RBC DISPLAY - SCAN5') NOOBJECT


              $PRINT(:LP:)  DEBUG

   1          SCAN5$MOD:  DO;

   2   1      HEIGHT: PROCEDURE(Y) ADDRESS EXTERNAL;
   3   2      DECLARE Y ADDRESS;
   4   2      END HEIGHT;

   5   1      DECLARE (K3,K4,K5) BYTE EXTERNAL;
   6   1      DECLARE BEIL(2) ADDRESS EXTERNAL, CEIL(2) ADDRESS EXTERNAL, CLOUD(2) ADDRESS EXTERNAL;
   7   1      DECLARE CLOUDY(22) ADDRESS EXTERNAL, HEIT(20) ADDRESS EXTERNAL, RCT(2) BYTE EXTERNAL;

   8   1      SCAN5: PROCEDURE(NO2) PUBLIC;

              /* THIS PROCEDURE TAKES THE MAXIMUM PEAK HEIGHT IN 1/4 DEGREES FOR 5 SCANS AND
              FINDS A REPRESENTATIVE CLOUD BASE HEIGHT(CBH) FOR 5 SCANS. THE FIVE HEIGHTS(IN
              1/4 DEGREES) ARE FIRST TEST TO SEE IF TWO VALUES ARE EQUAL. IF THERE ARE THIS
              VALUE IS CHOSEN AS THE CBH. IF THERE ARE MORE THAN ONE PAIR THAT ARE EQUAL, THE
              LOWER PAIR IS SELECTED. IF THE FIRST TEST FAILS, THE FIVE ARE TESTED TO SEE IF
              TWO HEIGHTS ARE WITHIN +-4 VALUES (+-1 DEGREE) OF EACH OTHER. AGAIN IF THERE ARE
              THE LOWER ONE OF THE MATCH IS SELECTED AS THE CBH. IF TWO PAIRS SATISFY THIS
              CRITERIA THE LOWER ONE OF THE LOWER PAIR IS SELECTED. IF BOTH METHODS FAIL A
              FINAL ATTEMPT IS MADE TO SEE IF ONE OF THE FIVE IS WITHIN +-6 VALUES (+-1 1/2 DEGREES)
              OF THE PREVIOUS CBH PROVIDED THAT THE PREVIOUS CBH WAS NOT "NO CBH". IF THERE ARE,
              THE LOWER OF THE TWO IS CHOSEN. IF MORE THAN ONE MATCHS, THEN THE LOWEST IS
              SELECTED.     */

   9   2      DECLARE (J,K,NO2,Q) BYTE;
  10   2      DECLARE (I,X1,X2,X3,X4,X5) ADDRESS;

  11   2      Q,CEIL(NO2) = 0;                /*Q IS IDENTIFIER TO SHOW WHEN A CBH HAS BEEN FOUND.
                                              CEIL(NO2) IS WHERE THE HEIGHT OF CBH WILL BE STORED WHEN FOUND*/
  12   2      K = 10 * NO2;                   /*K WILL SHOW WHERE THE CURRENT MAX PEAKS OF THE LAST 5 SCANS
                                              ARE FOUND IN HEIT*/

              /* ****************FIRST METHOD FIND TWO VALUES EQUAL*********************** */

  13   2      DO I = K TO K+3;                /*DO LOOP TO COMPARE VALUES FROM SCANS 1 THRU 4*/
  14   3      DO J = (I+1) TO K+4;            /*DO LOOP TO COMPARE VALUES FROM SCANS 2 THRU 5 WITH SCANS 1-4 ABOVE*/
  15   4      IF HEIT(I) < 349 THEN DO;       /*THE SCANS MUST HAVE A PEAK NOT A VALUE REPRESENTING NO CBH*/
  17   5      IF HEIT(I) = HEIT(J) THEN DO;   /*IF TWO HEIGHTS ARE EQUAL*/
  19   6      IF CEIL(NO2) = 0 THEN DO;       /*FIRST TIME ANY TWO VALUES ARE EQUAL*/
  21   7      CEIL(NO2) = HEIT(I);            /*STORE THE FOUND MATCH IN CEIL(NO2)*/
  22   7      Q = 1;                          /*Q=1 MEANS A MATCH HAS BEEN FOUND*/
  23   7      END;
  24   6      ELSE DO;                        /*TWO PAIRS OF DATA POINTS ARE EQUAL */
  25   7      IF CEIL(NO2) > HEIT(I) THEN DO;  /*SECOND PAIR OF DATA POINTS ARE AT A LOWER HEIGHT THAN FIRST*/
  27   8      CEIL(NO2) = HEIT(I);              /*REPLACE FIRST PAIR WITH SECOND PAIR, BECAUSE 2ND PAIR LOWER*/
  28   8      END;  END;  END;  END;  END;  END;

              /* ***** SECOND METHOD (FIRST METHOD FAILED) FIND TWO VALUES WITHIN +-4 VALUES (+-1 DEGREE) ***** */
```

```
 34   2   IF Q = 0 THEN DO;                /*TWO HEIGHTS NOT EQUAL-NOW CHECK TO FIND 2 HEIGHTS CLOSE TO EACH OTHER*/
 36   3   X2 = 0;                          /*DUMMY VARIABLE TO STORE ANY MATCHES FOUND*/
 37   3   DO I = K TO K+3;                 /*DO LOOP TO CHECK SCANS 1 THRU 4*/
 38   4   IF HEIT(I) < 349 THEN DO;        /*ONLY WANT TO COMPARE THOSE SCANS WITH PEAKS; NOT WITH NOCBH*/
 40   5   X5 = HEIT(I);                    /*X5 WILL BE THE CURRENT HEIGHT TO BE COMPARED */
 41   5   IF X5 < 32 THEN X4 = 27;         /*X4 WILL BE CURRENT HEIGHT MINUS FOUR VALUES EXCEPT LOWEST VALUE
                                           CAN BE ONLY 27*/
 43   5   ELSE X4 = X5 - 4;                /*HEIT(I) - 4 IN X4*/
 44   5   IF X5 > 343 THEN X3 = 348;       /*X3 WILL BE CURRENT HEIGHT PLUS FOUR VALUES EXCEPT HIGHEST VALUE
                                           CAN ONLY BE 348*/
 46   5   ELSE X3 = X5 + 4;                /*HEIT(I) + 4 IN X3*/
 47   5   DO J = (I+1) TO K+4;             /*CHECK CURRENT HEIT(I)+-4 AGAINST OTHER HEIGHTS FROM SCAN 2-5*/
 48   6   IF HEIT(J) < 349 THEN DO;        /*ONLY WANT TO COMPARE THOSE SCANS WITH PEAKS; NOT WITH NOCBH*/
 50   7   IF HEIT(J) >= X4 THEN DO;        /*GREATER THAN CURRENT HEIT(I)-4*/
 52   8   IF HEIT(J) <= X3 THEN DO;        /*LESS THAN CURRENT HEIT(I)+4*/
 54   9   IF HEIT(J) < HEIT(I) THEN X1 = HEIT(J);  /*A MATCH HAS BEEN FOUND NOW PUT LOWER HEIGHT IN X1*/
 56   9   ELSE X1 = HEIT(I);
 57   9   IF X2 = 0 THEN X2 = X1;          /*FIRST PAIR OF DATA TO MATCH SO PUT THE LOWER VALUE FOUND IN X2*/
 59   9   ELSE DO;                         /*SECOND PAIR OF DATA TO MATCH SO MUST CHOOSE LOWER ONE*/
 60  10   IF X1 < X2 THEN X2 = X1;         /*IF CURRENT MATCH IS LOWER THAN THE PREVIOUS ONE PUT IN X1*/
 62  10   END;
 63   9   CEIL(NO2) = X2;                  /*PUT LOWER MATCH FOUND IN CEIL(NO2)*/
 64   9   Q = 1;                           /*IDENTIFIER THAT A MATCH HAS BEEN FOUND*/
 65   9   END;  END;  END;  END;  END;  END;  END;

          /* ***** THIRD METHOD (FIRST TWO FAILED) TEST IF ONE VALUE WITHIN +-6 VALUES(+-1 1/2DEGRESS)
                                           OF PREVIOUS CBH ***** */

 72   2   IF Q = 0 THEN DO;                /*NOT EQUAL,2 VALUES NOT CLOSE, SO NOW TRY CLOSE TO LAST REPORTED CBH*/
 74   3   IF BEIL(NO2) < 349 THEN DO;      /*LAST MINUTE"S VALUE MUST BE A PEAK VALUE NOT A NOCBH*/
 76   4   X2 = 0;                          /*X2 WILL BE USED TO STORE LOWER OF MATCH FOUND*/
 77   4   DO J = K TO K+4;                 /*COMPARE 5 SCANS AGAINST LAST MINUTE"S VALUE*/
 78   5   IF HEIT(J) < 349 THEN DO;          /*THE SCAN BEING COMPARED MUST BE A PEAK VALUE NOT NOCBH*/
 80   6   X5 = HEIT(J);                      /*THE CURRENT VALUE BEING COMPARED IS IN X5*/
 81   6   IF X5 < 34 THEN X4 = 27;           /*X4 WILL BE CURRENT VALUE MINUS 6 WITH A MINIMUM OF 27*/
 83   6   ELSE X4 = X5 - 6;                  /*HEIT(J) - 6 IN X4*/
 84   6   IF X4 > 341 THEN X3 = 348;         /*X3 WILL BE CURRENT VALUE PLUS 6 WITH A MAXIMUM OF 348*/
 86   6   ELSE X3 = X5 + 6;                  /*HEIT(J) + 6 IN X3*/
 87   6   IF BEIL(NO2) >= X4 THEN DO;        /*PREVIOUS HEIGHT > HEIT(J) - 6*/
 89   7   IF BEIL(NO2) <= X3 THEN DO;        /*PREVIOUS HEIGHT <= HEIT(J) + 6*/
 91   8   IF BEIL(NO2) < HEIT(J) THEN X1 = BEIL(NO2);  /*LOWER HEIGHT IN X1*/
 93   8   ELSE X1 = HEIT(J);
 94   8   IF X2 = 0 THEN X2 = X1;            /*FIRST MATCH WITH PREVIOUS REPORTED CBH*/
 96   8   ELSE DO;                           /*SECOND MATCH WITH PREVIOUS REPORTED CBH*/
 97   9   IF X1 < X2 THEN X2 = X1;           /*SECOND MATCH WAS LOWER THAN THE FIRST*/
 99   9   END;
100   8   CEIL(NO2) = X2;
101   8   Q = 1;
102   8   END;  END;  END;  END;  END;  END;
108   2   IF CEIL(NO2) = 0 THEN CEIL(NO2) = 1000;
          /*CEIL IS FROM 27 TO 348 FOR CLOUDS, 1000 FOR NOCIO*/

          /*NOW THAT A CBH HAS BEEN FOUND THE FOLLOWING CONVERTS FROM 1/4DEGREES TO FEET*/

110   2   IF CEIL(K3) > 348 THEN DO;                  /*CEIL FOUND IS NO CBH OR A MALFUNCTION*/
112   3   IF CEIL(K3) = 1000 THEN CLOUD(K3) = 10000;  /*CEIL FOUND IS NO CBH REPRESENTED BY 10000*/
```

```
PL/M-80 COMPILER    RBC DISPLAY - SCAN5

                                                          /*A MALFUNCTION SOMEWHERE*/
114  3     ELSE CLOUD(K3) = 0;
115  3     END;                                           /*A CBH HEIGHT HAS BEEN FOUND NOW CONVERT TO FEET*/
116  2     ELSE CLOUD(K3) = HEIGHT(CEIL(K3));             /*SAVE THE CURRENT CBH IN BEIL TO BE USED
117  2     BEIL(K3) = CEIL(K3);                           AS PREVIOUS READING IN NEXT DETERMINATION*/
                                                          /*DISABLE SO WE CAN STORE ALL CURRENT MAX,2ND MAX,
118  2     DISABLE;                                       AND CBH IN CASE LATER WE WANT TO SEND OUT*/
                                                          /*K4 WILL TELL WHERE CURRENT VALUESSTORED IN HEIT*/
119  2     K4 = K3 * 10;                                  /*K5 WILL SHOW WHERE CURRENT VALUESSTORED IN CLOUDY*/
120  2     K5 = K3 * 11;                                  /*FIVE SCAN VALUE FIRST IN DATA*/
121  2     CLOUDY(K5) = CLOUD(K3);
122  2     K5 = K5 + 1;                                   /*DO LOOP TO STORE MAX AND 2ND MAX PEAKS' HEIGHTS*/
123  2     DO I = 0 TO 9;
124  3     IF HEIT(K4+I) = 1000 THEN CLOUDY(K5+I) = 10000;  /*CONVERT 1000 INTO NOCBH*/
126  3     ELSE CLOUDY(K5+I) = HEIGHT(HEIT(K4+I));        /*CONVERT 1/4DEGREE HEIGHTS TO FEETHEIGHTS*/
127  3     END;
128  2     ENABLE;                                        /*SET SCAN COUNTER BACK TO FOUR*/
129  2     RCT(K3) = 4;                                   /*DO LOOP TO DISCARD OLDEST VALUE AND MAKE
130  2     DO I = 0 TO 3;                                 ROOM FOR NEXT VALUE*/
                                                          /*FOR MAX PEAK*/
131  3     HEIT(K4+I) = HEIT(K4+I+1);                     /*FOR SECONDARY MAX PEAKS*/
132  3     HEIT(K4+I+5) = HEIT(K4+I+6);
133  3     END;
134  2     END SCAN5;
135  1     END SCAN5$MOD;

MODULE INFORMATION:

     CODE AREA SIZE     = 0558H   1368D
     VARIABLE AREA SIZE = 0010H     16D
     MAXIMUM STACK SIZE = 0004H      4D
     132 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER    RBC DISPLAY - VARY

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE VARYMOD
OBJECT MODULE PLACED IN :F1:VARY.OBJ
COMPILER INVOKED BY:  PLM80 :F1:VARY.SRC TITLE('RBC DISPLAY - VARY')


              $PRINT(:LP:)  DEBUG

  1           VARY$MOD:  DO;

  2    1      DECLARE CTR ADDRESS EXTERNAL;

  3    1      VARY: PROCEDURE PUBLIC;

              /*THIS PROCEDURE CAUSES THE CURSOR TO MOVE VERY SLOWLY FOR THE FIRST CHANGE
              BUT THEN AFTER THE FIRST, THE CURSOR WILL MOVE VERY RAPIDLY*

  4    2      DECLARE I BYTE;

  5    2      IF CTR < 1 THEN DO;       /*CHECK TO SEE IF FIRST MOVE OF CURSOR OR A SUCCEEDING ONE*/
  7    3      DO I = 0 TO 19;           /*IF FIRST MOVE WAIT 1/2 SECOND BEFORE GOING ON*/
  8    4      CALL TIME(250);           /*JUST KILLING TIME*/
  9    4      END;
 10    3      CTR = CTR + 1;            /*INCREMENT MOVE COUNTER*/
 11    3      END;
 12    2      ELSE CALL TIME(200);      /*WASTE ONLY 20 MSEC BEFORE PROCEEDING FOR ANY MOVE AFTER FIRST ONE*/
 13    2      END VARY;
 14    1      END VARY$MOD;


MODULE INFORMATION:

     CODE AREA SIZE     = 0035H     53D
     VARIABLE AREA SIZE = 0001H      1D
     MAXIMUM STACK SIZE = 0002H      2D
     22 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

Appendix C

Conversion Table from Pulse Counts to Heights

Elevation Angle	Pulse Count	Height (ft)	Elevation Angle	Pulse Count	Height (ft)
6.75	27	47	14.00	56	100
7.00	28	49	14.25	57	102
7.25	29	51	14.50	58	103
7.50	30	53	14.75	59	105
7.75	31	54	15.00	60	107
8.00	32	56	15.25	61	109
8.25	33	58	15.50	62	111
8.50	34	60	15.75	63	113
8.75	35	62	16.00	64	115
9.00	36	63	16.25	65	116
9.25	37	65	16.50	66	118
9.50	38	67	16.75	67	120
9.75	39	69	17.00	68	122
10.00	40	71	17.25	69	124
10.25	41	72	17.50	70	126
10.50	42	74	17.75	71	128
10.75	43	76	18.00	72	130
11.00	44	78	18.25	73	132
11.25	45	80	18.50	74	134
11.50	46	81	18.75	75	136
11.75	47	83	19.00	76	138
12.00	48	85	19.25	77	140
12.25	49	87	19.50	78	142
12.50	50	89	19.75	79	144
12.75	51	91	20.00	80	147
13.00	52	92	20.25	81	148
13.25	53	94	20.50	82	150
13.50	54	96	20.75	83	152
13.75	55	98	21.00	84	154

Elevation Angle	Pulse Count	Height (ft)	Elevation Angle	Pulse Count	Height (ft)
21.25	85	156	35.25	141	283
21.50	86	158	35.50	142	285
21.75	87	160	35.75	143	288
22.00	88	162	36.00	144	291
22.25	89	164	36.25	145	293
22.50	90	166	36.50	146	296
22.75	91	168	36.75	147	299
23.00	92	170	37.00	148	302
23.25	93	172	37.25	149	304
23.50	94	174	37.50	150	307
23.75	95	176	37.75	151	310
24.00	96	178	38.00	152	313
24.25	97	180	38.25	153	315
24.50	98	182	38.50	154	318
24.75	99	184	38.75	155	321
25.00	100	187	39.00	156	324
25.25	101	189	39.25	157	327
25.50	102	191	39.50	158	330
25.75	103	193	39.75	159	333
26.00	104	195	40.00	160	336
26.25	105	197	40.25	161	339
26.50	106	199	40.50	162	342
26.75	107	202	40.75	163	345
27.00	108	204	41.00	164	348
27.25	109	206	41.25	165	351
27.50	110	208	41.50	166	354
27.75	111	210	41.75	167	357
28.00	112	213	42.00	168	360
28.25	113	215	42.25	169	363
28.50	114	217	42.50	170	367
28.75	115	219	42.75	171	370
29.00	116	222	43.00	172	373
29.25	117	224	43.25	173	376
29.50	118	226	43.50	174	380
29.75	119	229	43.75	175	383
30.00	120	231	44.00	176	386
30.25	121	233	44.25	177	390
30.50	122	235	44.50	178	393
30.75	123	238	44.75	179	397
31.00	124	240	45.00	180	400
31.25	125	243	45.25	181	404
31.50	126	245	45.50	182	407
31.75	127	248	45.75	183	411
32.00	128	250	46.00	184	414
32.25	129	252	46.25	185	418
32.50	130	255	46.50	186	422
32.75	131	257	46.75	187	425
33.00	132	260	47.00	188	429
33.25	133	262	47.25	189	433
33.50	134	265	47.50	190	437
33.75	135	267	47.75	191	440
34.00	136	270	48.00	192	444
34.25	137	272	48.25	193	448
34.50	138	275	48.50	194	452
34.75	139	278	48.75	195	456
35.00	140	280	49.00	196	460

Elevation Angle	Pulse Count	Height (ft)	Elevation Angle	Pulse Count	Height (ft)
49.25	197	464	63.25	253	794
49.50	198	468	63.50	254	802
49.75	199	472	63.75	255	811
50.00	200	478	64.00	256	820
50.25	201	481	64.25	257	829
50.50	202	485	64.50	258	839
50.75	203	490	64.75	259	848
51.00	204	494	65.00	260	858
51.25	205	498	65.25	261	868
51.50	206	503	65.50	262	878
51.75	207	507	65.75	263	888
52.00	208	512	66.00	264	898
52.25	209	517	66.25	265	909
52.50	210	521	66.50	266	920
52.75	211	526	66.75	267	931
53.00	212	531	67.00	268	942
53.25	213	536	67.25	269	954
53.50	214	541	67.50	270	966
53.75	215	546	67.75	271	978
54.00	216	551	68.00	272	990
54.25	217	556	68.25	273	1003
54.50	218	561	68.50	274	1015
54.75	219	566	68.75	275	1029
55.00	220	571	69.00	276	1042
55.25	221	577	69.25	277	1056
55.50	222	582	69.50	278	1070
55.75	223	587	69.75	279	1084
56.00	224	593	70.00	280	1099
56.25	225	599	70.25	281	1114
56.50	226	604	70.50	282	1130
56.75	227	610	70.75	283	1145
57.00	228	616	71.00	284	1162
57.25	229	622	71.25	285	1178
57.50	230	628	71.50	286	1195
57.75	231	634	71.75	287	1213
58.00	232	640	72.00	288	1231
58.25	233	644	72.25	289	1250
58.50	234	653	72.50	290	1269
58.75	235	659	72.75	291	1288
59.00	236	666	73.00	292	1308
59.25	237	672	73.25	293	1329
59.50	238	679	73.50	294	1350
59.75	239	686	73.75	295	1372
60.00	240	693	74.00	296	1395
60.25	241	700	74.25	297	1418
60.50	242	707	74.50	298	1442
60.75	243	714	74.75	299	1467
61.00	244	722	75.00	300	1493
61.25	245	729	75.25	301	1519
61.50	246	737	75.50	302	1547
61.75	247	744	75.75	303	1575
62.00	248	752	76.00	304	1604
62.25	249	760	76.25	305	1635
62.50	250	768	76.50	306	1666
62.75	251	777	76.75	307	1699
63.00	252	785	77.00	308	1733

Elevation Angle	Pulse Count	Height (ft)	Elevation Angle	Pulse Count	Height (ft)
77.25	309	1768	82.00	328	2846
77.50	310	1804	82.25	329	2939
77.75	311	1842	82.50	330	3038
78.00	312	1882	82.75	331	3144
78.25	313	1923	83.00	332	3259
78.50	314	1966	83.25	333	3380
78.75	315	2011	83.50	334	3510
79.00	316	2058	83.75	335	3652
79.25	317	2107	84.00	336	3806
79.50	318	2158	84.25	337	3972
79.75	319	2212	84.50	338	4155
80.00	320	2268	84.75	339	4353
80.25	321	2328	85.00	340	4572
80.50	322	2390	85.25	341	4814
80.75	323	2456	85.50	342	5082
81.00	324	2525	85.75	343	5383
81.25	325	2599	86.00	344	5720
81.50	326	2676	86.25	345	6103
81.75	327	2759	86.50	346	6540